

Über einen Automaten mit Speicherband

Pufferautomat

Der Technischen Fakultät
der Universität Erlangen-Nürnberg
zur Erlangung des Grades
DOKTOR-INGENIEUR

vorgelegt

von

Roland Vollmar

Erlangen 1969

1969

Über einen Automaten mit Speicherband

Pufferautomat

Der Technischen Fakultät
der Universität Erlangen-Nürnberg
zur Erlangung des Grades
DOKTOR-INGENIEUR

vorgelegt

von

Roland Vollmar

Erlangen 1969



Als Dissertation genehmigt von der
Technischen Fakultät

der Universität Erlangen-Nürnberg

Tag der Einreichung: 12.8.1968

Tag der Promotion: 13.12.1968

Dekan: Prof. Dr. B. Ilschner

Berichterstatter: Prof. Dr. W. Händler
Prof. Dr. B. Schlender

Übersicht

0. Einleitung	0
1. Definition des Pufferautomaten	2
2. Turingmaschinen und Pufferautomaten	10
3. Vergleich einiger Pufferautomatentypen	22
4. Einordnung der deterministischen Puffersprachen in die Typenhierarchie für formale Sprachen	48
5. Operationen mit deterministischen Puffersprachen	66
6. Pufferautomaten und kombinatorische Systeme	76
7. Entscheidbarkeitsfragen bei Puffersprachen	86
8. Eine Hierarchie von Pufferautomaten	98
Literatur	108



Digitized by the Internet Archive
in 2026

https://archive.org/details/IA41546207_0066

0. Einleitung

Die Untersuchung abstrakter Automaten mit Speicherbändern wurde angeregt von der Theorie der rekursiven Funktionen und der Mathematischen Linguistik.

Kellerautomaten, linear beschränkte Automaten und Turingmaschinen sind typische Automaten dieser Art. Im Unterschied zu den endlichen, erkennenden Automaten verfügen die genannten Typen über einen wachsenden, potentiell unendlichen Speicher, von dessen Inhalt sie ihre Vorgehensweise abhängig machen können. Ihre unterschiedlichen Fähigkeiten beim Annehmen von Klassen formaler Sprachen resultieren aus den verschiedenen Einschränkungen, denen sie beim Auslesen der Speicherinhalte unterworfen sind.

Nachdem man längere Zeit in der Hauptsache solche Automaten betrachtete, die in enger, wechselseitiger Beziehung zu den Sprachklassen der CHOMSKY-Hierarchie stehen, sind in den beiden letzten Jahren verschiedene Typen von Automaten untersucht worden, die sich nicht mehr in natürlicher Weise in diese Hierarchie einordnen lassen. Dies gilt auch für den hier eingeführten Automaten, der Pufferautomat genannt werden soll, und der aus zweierlei Gründen interessant scheint:

- 1) In naheliegender Analogie zum Kellerautomaten, bei dem der Zugriff auf das Speicherband nach dem Prinzip "was zuletzt eingelesen wurde, wird zuerst ausgelesen" erfolgt, wird ein Automat definiert, bei dem nach dem Grundsatz "was zuerst eingelesen wurde, wird zuerst ausgelesen" verfahren wird.
- 2) Das Annehmen von Ketten aus einer freien Halbgruppe über einem endlichen Alphabet, die aus zwei oder mehreren gleichen Teilketten durch Konkatenation entstehen, soll

in "einfacher Weise" möglich sein.

Wir werden im folgenden ausführlich auf das Grundmodell eingehen. Danach werden verschiedene Modifikationen, die sich aus dem Grundmodell durch Einschränkungen beim Einlesen von Symbolen vom Eingabeband, beim Einschreiben auf das Speicherband und beim Auslesen von diesem ergeben, betrachtet und in ihrem Verhalten zueinander und zu anderen Automatentypen verglichen. Der Vergleich erfolgt dabei aufgrund der von den betreffenden Automaten "angenommenen" Mengen von Ketten aus einer vorgegebenen freien Halbgruppe über einem endlichen Alphabet. In diesem Zusammenhang wird dann für ein spezielles Pufferautomatenmodell untersucht, wie sich die Klasse der von ihm angenommenen Sprachen in die Hierarchie der formalen Sprachen nach CHOMSKY einfügt. Diese Klasse wird weiterhin bezüglich ihres Verhaltens gegenüber gewissen Operationen, wie Vereinigungs- und Durchschnittsbildung, usw. betrachtet. Da neben der Annahme von Sprachen auch jeweils Möglichkeiten, sie zu erzeugen, von Interesse sind, wird ein Zusammenhang zwischen Pufferautomaten und speziellen kombinatorischen Systemen hergestellt. Ein weiterer Abschnitt ist Entscheidungsbarkeitsfragen bei diesen Sprachen gewidmet. Die Untersuchung von Automaten mit einer Vielzahl von Puffern schließt die Arbeit ab.

1. Definition des Pufferautomaten

In diesem Abschnitt wird der Pufferautomat definiert werden, und es werden in Anlehnung an die bei den anderen Automatentypen verwendete Notation (GINSBURG/ROSE (1966)) Schreibweisen eingeführt werden, die sein Verhalten zu charakterisieren gestatten.

Zunächst sei noch erklärt, in welchem Sinn der Ausdruck "Korrespondenz" gebraucht wird. Wir lehnen uns dabei an SCHMITT (1966) an, der dafür die Bezeichnung "Transformation" benutzt.

Def.: Ein Tripel $\delta = (G, A, B)$ von Mengen G, A und B heißt eine Korrespondenz, wenn gilt: $G \subset A \times B$.

G heißt dann die Relation von δ , A die Ausgangsmenge und B die Zielfmenge von δ . Um auszudrücken, daß δ die Ausgangsmenge A und die Zielfmenge B hat, schreibt man auch: $\delta : A \rightarrow B$.

Die Menge $\text{pr}_1\langle G \rangle$ heißt Definitionsbereich, $\text{pr}_2\langle G \rangle$ Wertebereich von δ .

Für $a \in A$ wird gesetzt:

$$\delta[a] := \{b \mid b \in B \wedge (a, b) \in G\}.$$

Zur weiteren Abkürzung der Schreibweise sei noch eingeführt: Ist $\delta = (G, A \times A', B)$, so schreibt man

$$\delta[(a, a')] = \delta[a, a'].$$

Die i-Projektion eines n -tupels $m = (m_1, m_2, \dots, m_n)$ ist dabei definiert als

$$\text{pr}_i[m] := \begin{cases} \{m_i\} & \text{falls } 1 \leq i \leq n \\ \emptyset & \text{sonst} \end{cases}$$

Für die i -Projektion einer Menge von n -tupeln gilt:

Ist $M \subset M_1 \times M_2 \times \dots \times M_n$, so ist definiert

$$\text{pr}_i\langle M \rangle := \bigcup_{m \in M} \text{pr}_i[m].$$

Außerdem werden später die Begriffe "eindeutige Korrespondenz"

und "endliche Korrespondenz" benötigt.

Def.: Eine Korrespondenz $\delta = (G, A, B)$ heißt eindeutig, wenn für alle $a \in A$ gilt: $\delta[a]$ enthält höchstens ein Element.

Def.: Eine Korrespondenz $\delta = (G, A, B)$ heißt endlich, wenn der Definitions- und der Wertebereich von δ endliche Mengen sind.

Als erster Typ werde der nichtdeterministische, unvollständige Pufferautomat eingeführt.

Def.: Ein nichtdeterministischer, unvollständiger Pufferautomat ist ein Septupel $\mathcal{A}^* = (X, Z, P, B_1, B_2, \delta^*, z_0)$.

Dabei sind X, Z und P endliche, nichtleere Mengen,

$B_1 := \{b_{10}, b_{11}\}$, $B_2 := \{b_{20}, b_{21}, \dots, b_{2m}\}$ mit $m \in \mathbb{N}$, $z_0 \in Z$, und δ^* ist eine endliche Korrespondenz von $Z \times X \times \Omega(P)$ in $Z \times B_1 \times \Omega(P) \times B_2$.¹⁾

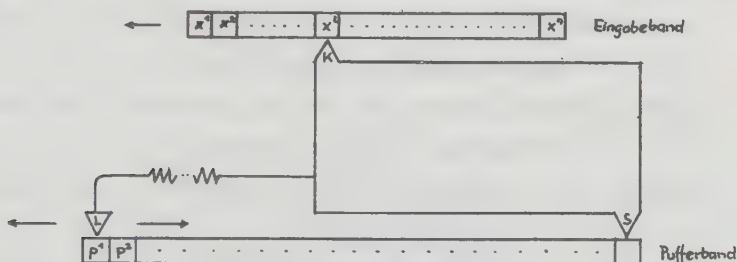
Folgende Sprechweise soll künftig verwandt werden:

X ist das Eingabealphabet, Z die Zustandsmenge, P das Pufferalphabet, B_1 die Menge, die die Eingabebandbewegung beschreibt, B_2 die Menge, die die Bewegung des Pufferlesekopfes angibt und z_0 der Anfangszustand des Automaten.

Der einfacheren Beschreibung wegen ist es wohl manchmal nützlich, von einer bildlichen Darstellung eines Pufferautomaten auszugehen, wie man ja auch mit dem Begriff "Turingmaschine" gewisse anschauliche Vorstellungen verbindet. Der Kern des Pufferautomaten ist ein endlicher Automat, der über einen Band-

¹⁾ Die freie Halbgruppe über einem endlichen Alphabet A mit der Konkatenation als Verknüpfung wird mit $\Omega(A)$ bezeichnet. Elemente einer freien Halbgruppe werden Wörter oder Ketten genannt, und das neutrale Element wird mit Λ bezeichnet. Die Menge aller endlich langen Wörter über einem endlichen Alphabet A werde ebenfalls mit $\Omega(A)$ bezeichnet. Aus dem Zusammenhang wird aber jeweils die Bedeutung klar werden.

lesekopf K von einem Eingabeband Symbole entgegennimmt. Das Eingabeband ist in einzelne Felder unterteilt, und jedes dieser Felder kann genau ein Symbol einer endlichen Kette aus Elementen von X aufnehmen. Zum Automaten gehört weiterhin ein Arbeitsband, das als Pufferband oder einfach als Puffer bezeichnet wird und das ebenfalls in Felder unterteilt ist, die jeweils genau ein Symbol des Pufferalphabets enthalten können. Das Pufferband wird über den Pufferschreibkopf S beschrieben. Es ist in seiner Länge nicht begrenzt. Der Pufferlesekopf L kann jeweils eine beliebige, aber endliche Anzahl von Symbolen am Beginn des Pufferbandes betrachten.



Die Beziehung $\delta^*[z_1, x_j, \pi_k] \supset \{(z_1, b_{1q}, \pi_r, b_{2s})\}$ mit $z_1, z_1 \in Z, x_j \in X, \pi_k, \pi_r \in \Omega(P), b_{1q} \in B_1$ und $b_{2s} \in B_2$ bedeutet dann: Befindet sich der Pufferautomat im Zustand z_1 , steht das Symbol x_j auf dem Feld des Eingabebandes, auf das der Bandlesekopf zeigt und enthalten die ersten Felder des Pufferbandes die aus Symbolen aus dem Pufferalphabet bestehende Kette π_k , so geht der Automat in den Zustand z_1 über, das Eingabeband wird um q Felder nach links bewegt und der Pufferlesekopf um s Felder des Pufferbandes (nach rechts) in Richtung zum Pufferschreibkopf, falls die Länge der auf dem Pufferband befind-

lichen Symbolkette π_r ist, und die Kette π_r wird auf das Pufferband geschrieben; waren vor dem Einschreiben der Kette π_r dagegen nur s Felder oder weniger beschrieben, so erfolgt ein vollständiges Löschen des Pufferbandes, d.h. der Pufferschreib- und der Pufferlesekopf stehen auf demselben Feld. Hervorgehoben werden soll, daß für die Reihenfolge des Einschreibens in den Puffer und das Auslesen aus dem Puffer folgendes gilt:

- (1) Zunächst bewegt sich der Pufferlesekopf um s Felder auf den Pufferschreibkopf zu, falls die Länge a des Puffers π_r $> s$ war und um a Felder, falls $a \leq s$ galt.
- (2) Dann wird eine Kette π_r z.B. der Länge b in den Puffer eingeschrieben und damit der Abstand zwischen Pufferlese- und Pufferschreibkopf um b Felder erhöht.

Das Nichtdeterminiertsein des Pufferautomaten bringt es mit sich, daß die Übergänge in andere Konfigurationen nicht eindeutig zu sein brauchen, so daß die obige Beschreibung nur eine Möglichkeit des Verhaltens aufzeigt.

Im Vergleich zum Kellerautomaten ist noch folgendes festzustellen:

- 1) Die Aufnahme eines ausgezeichneten Symbolen in das Pufferalphabet - analog zu dem im Kelleralphabet, das das Leersein des Kellers markiert - scheint überflüssig.
- 2) Für die Beschreibung der Eingabebandbewegung wurde eine andere Notation gewählt: Beim Kellerautomaten von GINSBURG (1966) wird auf die Einführung der Menge B_1 verzichtet, dafür aber als Eingabemenge $X \cup \{\Lambda\}$ gegeben und definiert, daß das Auftreten von Λ an der entsprechenden Stelle des Argumentes der Übergangskorrespondenz keine Bewegung des

Eingabebandes bewirkt.

Um das Verhalten von Pufferautomaten bequem beschreiben zu können, wollen wir noch einige Vereinbarungen treffen.

Für den obigen Automaten $\mathcal{A}^n$ werde über $Z \times \Omega(X) \times \Omega(P)$ die Relation $\vdash_{\mathcal{A}^n}^*$ definiert:

Def.: Seien $z, z' \in Z, x \in X, w \in \Omega(X), \pi_i, \pi_k, \pi_l, \pi_r, \pi_t, \pi \in \Omega(P)$,

$b_{10}, b_{11} \in B_1$ und $b_{25} \in B_2$.

Dann gelte $(z, xw, \pi_l \pi_i \pi) \vdash_{\mathcal{A}^n}^* (z', xw, \pi_l \pi \pi_r)$,

falls $(z', b_{10}, \pi_r, b_{25}) \in \delta^*[z, x, \pi_k]$

$\wedge \pi_k \neq \Lambda \wedge \pi_l \pi_i = \pi_k \wedge |\pi_i| = s$ 1)

$(z, xw, \pi_i \pi_l \pi) \vdash_{\mathcal{A}^n}^* (z', w, \pi_l \pi \pi_r)$,

falls $(z', b_{11}, \pi_r, b_{25}) \in \delta^*[z, x, \pi_k]$

$\wedge \pi_k \neq \Lambda \wedge \pi_l \pi_i = \pi_k \wedge |\pi_i| = s$

$(z, xw, \pi_k \pi_t \pi) \vdash_{\mathcal{A}^n}^* (z', xw, \pi \pi_r)$,

falls $(z', b_{10}, \pi_r, b_{25}) \in \delta^*[z, x, \pi_k] \wedge \pi_k \neq \Lambda \wedge |\pi_k \pi_t| = s \neq 0$

$(z, xw, \pi_k \pi_t \pi) \vdash_{\mathcal{A}^n}^* (z', w, \pi \pi_r)$,

falls $(z', b_{11}, \pi_r, b_{25}) \in \delta^*[z, x, \pi_k] \wedge \pi_k \neq \Lambda \wedge |\pi_k \pi_t| = s \neq 0$.

Für den Fall, daß $\pi_k = \Lambda$ ist, soll gelten:

$(z, xw, \Lambda) \vdash_{\mathcal{A}^n}^* (z', xw, \pi_r)$,

falls $(z', b_{10}, \pi_r, b_{25}) \in \delta^*[z, x, \Lambda]$

$(z, xw, \Lambda) \vdash_{\mathcal{A}^n}^* (z', w, \pi_r)$,

falls $(z', b_{11}, \pi_r, b_{25}) \in \delta^*[z, x, \Lambda]$.

Es gelte $(z, w, \pi) \vdash_{\mathcal{A}^n}^* (z', w', \pi')$,

falls $z = z^1, z^2, \dots, z^{k+1} = z' \in Z, w = w^1, w^2, \dots, w^k \in \Omega(X) \setminus \{\Lambda\}$,

$w^{k+1} = w' \in \Omega(X), \pi = \pi^1, \pi^2, \dots, \pi^{k+1} = \pi' \in \Omega(P)$ und ein end-

liches $k \in \mathbb{N}$ existieren, so daß für $1 \leq i \leq k$ gilt:

$(z^i, w^i, \pi^i) \vdash_{\mathcal{A}^n}^* (z^{i+1}, w^{i+1}, \pi^{i+1})$.

¹⁾ $|\pi|$ bezeichnet die Länge der Kette $\pi \in \Omega(P)$; insbesondere gilt $|\Lambda| = 0$, wobei Λ das neutrale Element der freien Halbgruppe ist.

Gilt $(z, w, \pi) \vdash_{\mathcal{Z}^*} (z', w', \pi')$, so sagt man, im Automaten $\mathcal{Z}^*$ findet ein direkter Übergang von (z, w, π) nach (z', w', π') statt.

Gilt $(z, w, \pi) \vdash_{\mathcal{Z}^*}^* (z', w', \pi')$, so sagt man, im Automaten $\mathcal{Z}^*$ findet ein Übergang von (z, w, π) nach (z', w', π') statt, oder, es gibt im Automaten $\mathcal{Z}^*$ eine Übergangsfolge, die (z, w, π) in (z', w', π') überführt.

Die Anzahl der direkten Übergänge wird als Länge der Übergangsfolge bezeichnet.

Unter den Gesichtspunkten, unter denen Automaten betrachtet und verglichen werden können, sei zunächst der des Erkennens von bestimmten Mengen herausgegriffen. Solche erkennenden Automaten sollen feststellen, ob Ketten aus einer freien Halbgruppe über einem Alphabet zu bestimmten Untermengen dieser freien Halbgruppe gehören.

Def.: Sei $\mathcal{Z}^*$ der oben angegebene Pufferautomat.

Ein Wort $w \in \Omega(X)$ heiße vom Pufferautomaten $\mathcal{Z}^*$ angenommen, wenn gilt: Es gibt in $\mathcal{Z}^*$ eine Übergangsfolge

$$(z_0, w, \Lambda) \vdash_{\mathcal{Z}^*}^* (z, \Lambda, \Lambda) \text{ mit } z \in Z.$$

Die Menge der vom Pufferautomaten $\mathcal{Z}^*$ angenommenen Wörter $w \in \Omega(X)$ werde mit $T(\mathcal{Z}^*)$ bezeichnet

Eine Kette w von Symbolen aus X gilt also dann als vom Automaten $\mathcal{Z}^*$ angenommen, wenn der Pufferautomat aus dem Anfangszustand z_0 und der Kette w auf dem Eingabeband und leerem Puffer durch eine endliche Übergangsfolge in eine Konfiguration gelangt, in der bei abgearbeitetem Eingabeband der Puffer leer ist. Bezüglich des dabei angenommenen Zustandes sollen keine Einschränkungen getroffen werden. Wie dies bei nichtdeterministi-

schen endlichen erkennenden Automaten üblich ist, soll auch hier gelten, daß mindestens eine aus der u.U. großen Zahl von Übergangsfolgen obiger Bedingung genügen muß.

Um Automaten vergleichen zu können, führen wir die Definitionen des Äquivalentseins und der Verwandtschaft von Automaten ein, und zwar für alle solche Automaten, für die wir das Annehmen von Mengen definieren.

Def.: Zwei Automaten α und α' heißen zueinander äquivalent, wenn sie die gleichen Mengen annehmen, d.h. wenn gilt:

$$T(\alpha) = T(\alpha').$$

Außerdem wollen wir noch eine etwas schwächere Forderung als die der Äquivalenz von Automaten einführen:

Def.: Ein Automat α' heißt verwandt mit einem Automaten α genau dann, wenn es eine feste, endliche, evtl. leere, Kette φ von Symbolen aus einem Alphabet, das durchschnittsfremd mit dem Eingabealphabet von α ist, gibt, so daß gilt:

$$u \in T(\alpha) \Leftrightarrow u\varphi \in T(\alpha'), \text{ wobei } u \neq \Lambda.$$

Da ein Automat eine vorgelegte Kette entweder annimmt oder nicht annimmt, und es keine dritte Möglichkeit gibt, impliziert die letzte Bedingung auch folgendes: $v \notin T(\alpha) \Leftrightarrow v\varphi \notin T(\alpha')$. Man könnte die Bedingung in der obigen Definition auch durch die letzte Formulierung ersetzen.

Angemerkt werden soll, daß es durchaus Elemente v' in $T(\alpha')$ geben kann, die nicht die Form $v\varphi$ besitzen. Dazu braucht es in $T(\alpha)$ keine Entsprechungen zu geben.

Die Relation "verwandt" stellt keine Äquivalenzrelation dar. Sie ist wohl reflexiv und transitiv, nicht aber symmetrisch. Man sieht leicht, daß gilt:

Zwei Automaten α und α' sind genau dann zueinander äquivalent, wenn α mit α' und α' mit α verwandt ist.

2. Turingmaschinen und Pufferautomaten

Die beherrschende Rolle, die die Turingmaschine in der Theorie der Berechenbarkeit und Entscheidbarkeit spielt, legt es nahe, andere abstrakte Automaten zunächst mit ihr zu vergleichen. Wir wollen deshalb eine deterministische Variante des im vorigen Abschnitt definierten Pufferautomaten, die zudem noch einige Einschränkungen aufweist - zumindest der Definition nach -, in Beziehung setzen zu Turingmaschinen.

Def.: Ein deterministischer, unvollständiger Pufferautomat ist ein Septupel $\mathcal{A} = (X, Z, P, B_1, B_2, \delta', z_0)$.

Dabei sind X, Z und P endliche, nichtleere Mengen,

$B_1 := \{b_{10}, b_{11}\}$, $B_2 := \{b_{20}, b_{21}, \dots, b_{2m}\}$ mit $m \in \mathbb{N}$, $z_0 \in Z$,

und δ' ist eine endliche eindeutige Korrespondenz von

$Z \times X \times \Omega(P)$ in $Z \times B_1 \times \Omega(P) \times B_2$.

Der Unterschied zum vorher eingeführten nichtdeterministischen Pufferautomaten liegt darin, daß die Korrespondenz δ' eindeutig sein muß, d.h., daß die Korrespondenz für jedes Argument höchstens einen Wert hat. Der deterministische Pufferautomat bleibt beim Erreichen einer bestimmten Konfiguration entweder stehen, wenn die entsprechende Korrespondenz die leere Menge als Wert hat, oder er geht eindeutig in eine Folgekonfiguration über.

Die Definitionen des direkten Übergangs, der Übergangsfolge, des Annehmens von Wörtern und Mengen für deterministische Pufferautomaten sind wörtlich die gleichen wie im Fall des nichtdeterministischen Pufferautomaten.

Von Turingmaschinen wollen wir im Sinn der folgenden Definition, die in Anlehnung an HERMES (1961) gegeben wird, sprechen.

Def.: Eine Turingmaschine $\mathcal{T}$ über $X \setminus \{x_0\}$ ist ein Quadrupel (X, Z, δ, z_1) . Dabei sind $X := \{x_0, x_1, \dots, x_m\}$ ($m \geq 1$) und $Z := \{z_1, \dots, z_n\}$ ($n \geq 1$) endliche, nichtleere Mengen, δ ist eine Abbildung von $Z \times X$ in $(X \cup \{l, r, s\}) \times Z$, wobei $l, r, s \notin X$ und $z_1 \in Z$. Mit x_0 sei das sog. uneigentliche Symbol bezeichnet. Wie üblich sei die Abbildung durch eine vierspaltige und $n(m+1)$ -reihige Tafel der Form

z_1	x_0	$\bar{x}^1$	z^1
$\dots$	$\dots$	$\dots$	$\dots$
z_1	x_m	$\bar{x}^{m+1}$	z^{m+1}
z_2	x_0	$\bar{x}^{m+2}$	z^{m+2}
$\dots$	$\dots$	$\dots$	$\dots$
z_n	x_m	$\bar{x}^{nm+n}$	z^{nm+n}

mit $\bar{x}^i \in X \cup \{l, r, s\}$ und $z^j \in Z$ ($i, j = 1, \dots, nm+n$) gegeben.

Ähnlich wie beim Pufferautomaten wollen wir auch hier eine besondere Schreibweise einführen, um das Arbeiten der Turingmaschine bequem darstellen zu können.

Def.: Gegeben sei die obige Turingmaschine $\mathcal{T}$.

Über $\Omega(X)Z(\Omega(X) \setminus \{\Lambda\})$ sei folgendermaßen eine Relation $\vdash_{\mathcal{T}}^*$ definiert:

Seien $z, z' \in Z, t, x, y \in X, u, v \in \Omega(X)$.

Dann gelte

$(uzyv) \vdash_{\mathcal{T}} (uz'tv)$	falls $\delta(z, y) = (t, z')$,
$(uzxyv) \vdash_{\mathcal{T}} (uxz'yv)$	falls $\delta(z, x) = (r, z')$,
$(uzy) \vdash_{\mathcal{T}} (uyz'x_0)$	falls $\delta(z, y) = (r, z')$,
$(uyztv) \vdash_{\mathcal{T}} (uz'y'tv)$	falls $\delta(z, t) = (l, z')$,
$(zxv) \vdash_{\mathcal{T}} (z'x_0xv)$	falls $\delta(z, x) = (l, z')$,
$(uzyv) \vdash_{\mathcal{T}} (uzyv)$	falls $\delta(z, y) = (s, z')$ (sic!).

Gilt $(uzv) \vdash_{\gamma} (u'z'v')$ mit $u, u' \in \Omega(X), v, v' \in \Omega(X) \setminus \{\Lambda\}$, so sagt man, in γ findet ein direkter Übergang von (uzv) nach $(u'z'v')$ statt.

Für $(uzv), (u'z'v') \in \Omega(X)Z(\Omega(X) \setminus \{\Lambda\})$ gelte:

$(uzv) \vdash_{\gamma}^* (u'z'v')$, falls $(uzv) = (u^1z^1v^1), (u^kz^kv^k), \dots$

$\dots, (u^{k+1}z^{k+1}v^{k+1}) = (u'z'v')$ existieren, so daß gilt:

$$(u^i z^i v^i) \vdash_{\gamma} (u^{i+1} z^{i+1} v^{i+1}) \quad \text{für } 1 \leq i \leq k.$$

Man sagt dann, es gibt in γ eine Übergangsfolge, die (uzv) in $(u'z'v')$ überführt, und ihre Länge wird als k definiert.

Die Definition einer Konfiguration wird anders als bei HERMES gegeben.

Def.: Mit Konfiguration werde jedes Element $(uzv) \in \Omega(X)Z(\Omega(X) \setminus \{\Lambda\})$ bezeichnet.

Eine Konfiguration $(uzxv) \in \Omega(X)Z(\Omega(X) \setminus \{\Lambda\})$ heißt

Endkonfiguration der Turingmaschine γ , falls $\delta(z, x) = (s, z')$!

Mit diesen Begriffen können wir den folgenden Satz angeben: ¹⁾

Satz 2-1: Zu jeder Turingmaschine γ über $X \setminus \{x_0\}$ gibt es einen deterministischen, unvollständigen Pufferautomaten $\mathcal{S}$, so daß gilt:

(1) Wird γ auf das erste Symbol eines Wortes

$w \in \Omega(X \setminus \{x_0\}) \setminus \{\Lambda\}$, das auf dem sonst leeren Band steht, angesetzt und gelangt γ durch eine Übergangsfolge endlicher Länge in eine Endkonfiguration, so wird $w\#$ mit $\# \notin X$ vom Pufferautomaten $\mathcal{S}$ angenommen und umgekehrt.

(2) Gelangt γ unter den obigen Bedingungen durch keine Übergangsfolge endlicher Länge in eine

¹⁾ Inzwischen wurde festgestellt, daß SHEPHERDSON und STURGIS (1963) von einer nach demselben Prinzip wie der Pufferautomat arbeitenden Maschine gezeigt haben, daß sie alle partiell-rekursiven Funktionen berechnet.

Endkonfiguration, so wird $w\#$ vom Pufferautomaten $\mathcal{S}'$ nicht angenommen und umgekehrt.

Bew.: Zur Turingmaschine $\mathcal{T}$ wird ein Pufferautomat $\mathcal{S}'$ so konstruiert, daß er das Verhalten von $\mathcal{T}$ simuliert: Zunächst wird die Teilkette w der Eingabebandsinschrift $w\#$ in den Puffer geschrieben, wobei ein besonderes Symbol p_A am Anfang hinzugefügt wird. Der Pufferinhalt wird dann in Abhängigkeit von der Änderung des Bandes der Turingmaschine und ihres Lese-Schreibkopfes modifiziert. Gelangt die Turingmaschine in eine Endkonfiguration, so wird der Pufferinhalt von $\mathcal{S}'$ gelöscht und das Eingabesymbol $\#$ gelesen, so daß es in $\mathcal{S}'$ einen Übergang nach $(z, \wedge, \wedge)$ mit $z \in Z$ gibt, d.h., $w\#$ wird vom Pufferautomaten $\mathcal{S}'$ angenommen. Hält die Turingmaschine nicht an, so gibt es in $\mathcal{S}'$ keine endliche Übergangsfolge, die $(z, \wedge, \wedge)$ erreicht.

Es sei eine Turingmaschine $\mathcal{T} = (X; Z; \delta, z_1)$ über $X \setminus \{x_0\}$ mit $X := \{x_0, x_1, \dots, x_m\}$ ($m \geq 1$), $Z := \{z_1, \dots, z_n\}$ ($n \geq 1$), $z_1 \in Z$ und der Abbildung δ gegeben.

Dann werden wir zeigen, daß der folgendermaßen konstruierte deterministische, unvollständige Pufferautomat $\mathcal{S}'$ obigen Bedingungen genügt: $\mathcal{S}' = (X', Z', P, B_1, B_2, \delta', z_0)$, wobei

$X' := X \cup \{\#\}$ mit $\# \notin X$, $Z' := Z \cup \{z_0\} \cup \{z_{ij}^l / i \in \{1, \dots, n\}, j \in \{1, \dots, m+1\} \wedge l \in \{1, 2\} \cup \{\bar{z}_{ij}^k / i \in \{1, \dots, n\} \wedge j \in \{1, \dots, m+1\} \wedge k \in \{0, 1, \dots, m+1\}\}$, wobei diese Mengen paarweise durch-

schnittsfremd sein sollen, $P := X \cup \{p_A, p_B, p_C\}$ mit $p_A, p_B, p_C \notin X$, $B_1 := \{b_{10}, b_{11}\}$, $B_2 := \{b_{20}, b_{21}\}$, $z_0 \in Z'$ und δ' eine endliche, eindeutige Korrespondenz von $Z' \times X' \times \bar{P}$ in $Z' \times B_1 \times \Omega(P) \times B_2$ ¹

¹) Künftig soll gelten: $\bar{P} := P \cup \{\wedge\}$; entsprechendes gilt auch für andere Mengen, nicht aber für Ketten, o.dgl.

ist, die im einzelnen folgendermaßen in Abhängigkeit von δ konstruiert wird:

1) Die Teilkette w wird folgendermaßen in den Puffer eingeschrieben: -Es befindet sich mindestens ein Symbol $\neq x_0$ auf dem Band der Turingmaschine und also auch auf dem Eingabeband von $\mathcal{S}'$.-

$$(\forall x / x \in X' \setminus \{\#\}) (\delta'[z_0, x, \wedge] = \{(z_0, b_{11}, p_A x, b_{20})\})$$

$$(\forall x / x \in X' \setminus \{\#\}) (\forall p / p \in P) (\delta'[z_0, x, p] = \{(z_0, b_{11}, x, b_{20})\})$$

$$\delta'[z_0, \#, p_A] = \{(z_1, b_{10}, p_A, b_{21})\}$$

Wir werden jetzt angeben, wie die verschiedenen Typen von Relationen aus der Tafel von $\mathcal{T}$ im Pufferautomaten $\mathcal{S}'$ simuliert werden müssen. Die Simulation jeder Relation geschieht in $\mathcal{S}'$ über Zwischenzustände z_{ij}^l , deren Indizierung von dem Zustand z_i ($i \in \{1, \dots, n\}$) und dem gerade unter dem Lese-Schreibkopf befindlichen Symbol x_j ($j \in \{0, \dots, m\}$) abhängig ist, um so die Eindeutigkeit der Korrespondenz δ' zu erreichen.

2) Für jede Relation der Form $\delta(z_i, x_j) = (y, z_k)$ in $\mathcal{T}$ muß in $\mathcal{S}'$ eingeführt werden:

$$\delta'[z_i, \#, x_j] = \{(z_{ij}^1, b_{10}, p_B y, b_{21})\}$$

$$(\forall p / p \in P \setminus \{p_B\}) (\delta'[z_{ij}^1, \#, p] = \{(z_{ij}^1, b_{10}, p, b_{21})\})$$

$$\delta'[z_{ij}^1, \#, p_B] = \{(z_k, b_{10}, \wedge, b_{21})\}$$

3) Für jede Relation der Form $\delta(z_i, x_j) = (r, z_k)$ in $\mathcal{T}$ muß in $\mathcal{S}'$ eingeführt werden:

$$\delta'[z_i, \#, x_j] = \{(z_{ij}^1, b_{10}, x_j p_B, b_{21})\}$$

$$\delta'[z_{ij}^1, \#, p_A] = \{(z_{ij}^2, b_{10}, x_0 p_A, b_{21})\}$$

$$(\forall p / p \in P \setminus \{p_A\}) (\delta'[z_{ij}^2, \#, p] = \{(z_{ij}^2, b_{10}, p, b_{21})\})$$

$$(\forall p / p \in P \setminus \{p_B\}) (\delta'[z_{ij}^2, \#, p] = \{(z_{ij}^2, b_{10}, p, b_{21})\})$$

$$\delta'[z_{ij}^2, \#, p_B] = \{(z_k, b_{10}, \wedge, b_{21})\}$$

4) Relationen der Form $\delta(z_i, x_j) = (1, z_k)$, durch die eine Bewegung des Lese-Schreibkopfes um ein Feld nach links verursacht wird, sind in $\mathcal{S}'$ etwas schwieriger zu simulieren. Durch besondere Zustände wird jeweils das gerade gelesene vorderste Puffersymbol "ins Gedächtnis aufgenommen", um an der entsprechenden Stelle die Anfangsmarkierung davor schreiben zu können.

Der Anschaulichkeit halber wollen wir die zu δ' hinzuzufügenden Relationen weitgehend einzeln aufschreiben.

$$\delta'[z_i, \#, x_j] = \{(z_{ij}^1, b_{40}, p_B x_j, b_{24})\}$$

$$\delta'[z_i^1, \#, x_0] = \{(\bar{z}_{ij}^0, b_{40}, \wedge, b_{24})\}$$

$$\delta'[z_{ij}^1, \#, x_1] = \{(\bar{z}_{ij}^1, b_{40}, \wedge, b_{24})\}$$

$$\delta'[z_{ij}^1, \#, x_m] = \{(\bar{z}_{ij}^m, b_{40}, \wedge, b_{24})\}$$

$$\delta'[z_{ij}^1, \#, p_A] = \{(\bar{z}_{ij}^{m+1}, b_{40}, \wedge, b_{24})\}$$

$$\delta'[\bar{z}_{ij}^0, \#, x_0] = \{(\bar{z}_{ij}^0, b_{40}, x_0, b_{24})\}$$

$$\delta'[\bar{z}_{ij}^0, \#, x_1] = \{(\bar{z}_{ij}^1, b_{40}, x_0, b_{24})\}$$

.

$$\delta'[\bar{z}_{ij}^0, \#, x_m] = \{(\bar{z}_{ij}^m, b_{40}, x_0, b_{24})\}$$

$$\delta'[\bar{z}_{ij}^0, \#, p_A] = \{(\bar{z}_{ij}^{m+1}, b_{40}, x_0, b_{24})\}$$

$$\delta'[\bar{z}_{ij}^0, \#, p_B] = \{(z_{ij}^1, b_{40}, p_C x_0, b_{24})\}$$

$$\delta'[\bar{z}_{ij}^1, \#, x_0] = \{(\bar{z}_{ij}^0, b_{40}, x_1, b_{24})\}$$

$$\delta'[\bar{z}_{ij}^1, \#, x_m] = \{(\bar{z}_{ij}^m, b_{40}, x_1, b_{24})\}$$

$$\delta'[\bar{z}_{ij}^1, \#, p_A] = \{(\bar{z}_{ij}^{m+1}, b_{40}, x_1, b_{24})\}$$

$$\delta'[\bar{z}_{ij}^1, \#, p_B] = \{(z_{ij}^1, b_{40}, p_C x_1, b_{24})\}$$

.
. . .

$$\delta'[\bar{z}_{ij}^m, \#, x_0] = \{(\bar{z}_{ij}^0, b_{40}, x_m, b_{24})\}$$

$$\delta'[\bar{z}_{ij}^m, \#, x_m] = \{(\bar{z}_{ij}^m, b_{40}, x_m, b_{24})\}$$

$$\delta'[\bar{z}_{ij}^m, \#, p_A] = \{(\bar{z}_{ij}^{m+1}, b_{40}, x_m, b_{21})\}$$

$$\delta'[\bar{z}_{ij}^m, \#, p_B] = \{(z_{ij}^2, b_{40}, p_C x_m, b_{21})\}$$

$$\delta'[\bar{z}_{ij}^{m+1}, \#, x_0] = \{(\bar{z}_{ij}^0, b_{40}, p_A, b_{21})\}$$

$$\delta'[\bar{z}_{ij}^{m+1}, \#, x_m] = \{(\bar{z}_{ij}^m, b_{40}, p_A, b_{21})\}$$

$$\delta'[\bar{z}_{ij}^{m+1}, \#, p_B] = \{(z_{ij}^2, b_{40}, p_A p_C x_0, b_{21})\}$$

$$(\forall p/p \in P \setminus \{p_C\})(\delta'[z_{ij}^2, \#, p] = \{(z_{ij}^2, b_{40}, p, b_{21})\})$$

$$\delta'[z_{ij}^2, \#, p_C] = \{(z_k, b_{40}, \wedge, b_{21})\}$$

5) Für jede Relation der Form $\delta(z_i, x_j) = (s, z_k)$ in $\mathcal{V}$ muß in $\mathcal{L}'$ eingeführt werden:

$$\delta'[z_i, \#, x_j] = \{(z_{ij}^1, b_{40}, \wedge, b_{21})\}$$

$$(\forall p/p \in P)(\delta'[z_{ij}^1, \#, p] = \{(z_{ij}^1, b_{40}, \wedge, b_{21})\})$$

$$\delta'[z_{ij}^1, \#, \wedge] = \{(z_i, b_{41}, \wedge, b_{20})\}$$

Der so konstruierte Automat zeigt folgendes Verhalten:

a). Zunächst sei die Wirkungsweise von $\mathcal{L}'$ beim Vorliegen einer Kette $w\#$, wobei $w \in \Omega(X) \setminus \{\wedge\}$, betrachtet.

Man sieht unmittelbar, daß mit 1) gilt:

$$(z_0, w\#, \wedge) \vdash_{\mathcal{L}'}^* (z_1, \#, wp_A) \text{ mit } z_0, z_1 \in Z'$$

b) Mit $u, v \in \Omega(X)$ gilt nach 2)

$$(z_i, \#, x_j vp_A u) \vdash_{\mathcal{L}'}^* (z_k, \#, yvp_A u)$$

c) Mit $u \in \Omega(X), v \in \Omega(X) \setminus \{\wedge\}$ gilt nach 3)

$$(z_i, \#, x_j vp_A u) \vdash_{\mathcal{L}'}^* (z_k, \#, vp_A ux_j),$$

$$\text{bzw. } (z_i, \#, x_j p_A u) \vdash_{\mathcal{L}'}^* (z_k, \#, x_0 p_A ux_j).$$

d) Mit $u, v \in \Omega(X)$ gilt nach 4)

$$(z_i, \#, x_j vp_A uy) \vdash_{\mathcal{L}'}^* (z_k, \#, yx_j vp_A u)$$

$$\text{bzw. } (z_i, \#, x_j vp_A) \vdash_{\mathcal{L}'}^* (z_k, \#, x_0 x_j vp_A)$$

e) Mit $u, v \in \Omega(X)$ gilt nach 5)

$$(z_i, \#, x_j vp_A u) \vdash_{\mathcal{L}'}^* (z_i, \wedge, \wedge).$$

Durch vollständige Induktion über die Länge der Übergangs-

folgen ergibt sich:

Zu jeder Übergangsfolge $(z, w) \vdash_7^* (uzv)$ mit $u \in \Omega(X)$ und $v \in \Omega(X) \setminus \{\lambda\}$, die die Länge k hat, gibt es genau eine Übergangsfolge in $\mathfrak{S}'$, die $(z, \#, wp_A)$ in $(z, \#, v p_A u)$ überführt und in der, abgesehen von Elementen aus $Z' \times \{\#\} \times \Omega(P)$ genau $k+1$ Elemente aus $Z \times \{\#\} \times \Omega(X \cup \{p_A\})$ auftreten. Außerdem gilt das Umgekehrte. Unter Anwendung von 1) und 5) folgt daraus die Richtigkeit der Behauptung (1).

Die Richtigkeit von (2) folgt durch Negieren von (1).

Damit ist gezeigt, daß es zu jeder Turingmaschine einen deterministischen, unvollständigen Pufferautomaten gibt, der ihre Vorgehensweise in dem oben formulierten Sinn zu simulieren gestattet. Diese Fähigkeit ergibt sich im wesentlichen daraus, daß das Eingabeband angehalten werden darf und auf diese Weise nach und nach alle schon im Puffer enthaltenen Informationen zugreifbar werden, ohne daß sie bei diesem Prozeß des Rundschiebens - dem Ringshift in Registern einer Rechenanlage entsprechend - verloren zu gehen brauchen.

Es ist anzunehmen, daß es zu jedem nichtdeterministischen, unvollständigen Pufferautomaten eine Turingmaschine gibt, die ihn in einer naheliegenden Weise zu simulieren gestattet. Wir wollen darauf im Augenblick jedoch nicht näher eingehen, sondern uns mit zwei Spezialisierungen des nichtdeterministischen, unvollständigen Pufferautomaten beschäftigen.

Zunächst wird ein Typ betrachtet, bei dem nur das Einschreiben einzelner Symbole in den Puffer möglich ist, während im zweiten Fall der Pufferautomat nur in Abhängigkeit einzelner Symbole am Beginn des Pufferbandes reagieren kann.

Lemma 2-1: Zu jedem nichtdeterministischen, unvollständigen

Pufferautomaten $\mathcal{A}^* = (X, Z, P, B_1, B_2, \delta^*, z_0)$, wobei

δ^* eine endliche Korrespondenz von $Z \times X \times \Omega(P)$ in $Z \times B_1 \times \Omega(P) \times B_2$ ist, gibt es einen nichtdeterministischen, unvollständigen Pufferautomaten

$\mathcal{A}' = (X, Z', P, B_1, B_2, \delta', z_0)$, wobei δ' eine endliche

Korrespondenz von $Z' \times X \times \Omega(P)$ in $Z' \times B_1 \times \bar{P} \times B_2$ ist, so daß gilt: $\mathcal{A}^*$ und $\mathcal{A}'$ sind zueinander äquivalent.

Bew.: Da die Korrespondenz δ^* endlich ist, läßt sich die größte Länge der dritten Komponente der Argumente von δ^* und die größte Länge der dritten Komponente der Werte von δ^* bestimmen; sie seien resp. r und s .

Dann wird gesetzt: $Z' := Z \cup \{(z, x, \pi, \bar{z}, b_i, \bar{\pi}, b_j, 1) \mid$

$(z, x, \pi, \bar{z}, b_i, \bar{\pi}, b_j, 1) \in Z \times X \times \Omega(P) \times Z \times B_1 \times \Omega(P) \times B_2 \times \{1, \dots, s-1\}$

$\wedge |\pi| \leq r \wedge |\bar{\pi}| \leq s\}$.

Z' bleibt so natürlich eine endliche Menge. Die Konstruktion von δ' ergibt sich folgendermaßen:

1) Gilt in $\mathcal{A}^*$ $\delta^*[z, x, \pi] \supset \{(\bar{z}, b_{4i}, p, b_{2q})\}$ mit $p \in \bar{P}$, so ist in $\mathcal{A}'$ aufzunehmen: $\delta'[z, x, \pi] \supset \{(\bar{z}, b_{4i}, p, b_{2q})\}$

2) Gilt in $\mathcal{A}^*$ $\delta^*[z, x, \pi] \supset \{(\bar{z}, b_{4i}, \bar{\pi}, b_{2q})\}$ mit $\bar{\pi} = p^1 \dots p^k \in \Omega(P)$ und $1 < k \leq s$, so ist in $\mathcal{A}'$ aufzunehmen

a) falls $k=2$

$\delta'[z, x, \pi] \supset \{((z, x, \pi, \bar{z}, b_{4i}, \bar{\pi}, b_{2q}, 1), b_{40}, p^1, b_{2q})\}$

$(\forall p/p \in \bar{P}) (\delta'[(z, x, \pi, \bar{z}, b_{4i}, \bar{\pi}, b_{2q}, 1), x, p] = \{(z, b_{4i}, p^2, b_{20})\})$

b) falls $k > 2$

$\delta'[z, x, \pi] \supset \{((z, x, \pi, \bar{z}, b_{4i}, \bar{\pi}, b_{2q}, 1), b_{40}, p^1, b_{2q})\}$

$(\forall p/p \in \bar{P}) (\forall j/1 \leq j \leq k-2) (\delta'[(z, x, \pi, \bar{z}, b_{4i}, \bar{\pi}, b_{2q}, j), x, p] = \{((z, x, \pi, \bar{z}, b_{4i}, \bar{\pi}, b_{2q}, j+1), b_{40}, p^{j+1}, b_{20})\})$

$$(\forall p/p \in \bar{P})(\delta'[(z, x, \pi, \bar{z}, b_{4i}, \bar{\pi}, b_{2q}, k-1), x, p] \\ = \{(\bar{z}, b_{4i}, p^k, b_{2q})\})$$

Durch die Wahl von Z' und die Konstruktion von δ' ist offensichtlich gewährleistet, daß bei Übergängen über die Zwischenzustände aus $Z' \setminus Z$ keine neuen "Verzweigungen" auftreten können.

Durch vollständige Induktion über die Länge der Übergangsfolgen ergibt sich dann sofort:

Gilt $(z, w, \pi) \vdash_{\mathcal{Z}'}^* (\bar{z}, w', \pi')$ mit $z, \bar{z} \in Z$, so gilt auch $(z, w, \pi) \vdash_{\mathcal{Z}}^* (\bar{z}, w', \pi')$ mit $z, \bar{z} \in \underline{Z}$ und umgekehrt.

Daraus folgt: $u \in T(\mathcal{Z}^*) \Leftrightarrow u \in T(\mathcal{Z}')$.

Lemma 2-2: Zu jedem nichtdeterministischen, unvollständigen

Pufferautomaten $\mathcal{Z}' = (X, Z'; P; B_1, B_2, \delta', z_0)$,

wobei δ' eine endliche Korrespondenz von

$Z' \times X \times \Omega(P')$ in $Z' \times B_1 \times \bar{P}' \times B_2$ ist, gibt es einen

nichtdeterministischen, unvollständigen Pufferauto-

maten $\mathcal{Z} = (X, Z, P, B_1, B_2, \delta, z_0)$, wobei δ eine endli-

che Korrespondenz von $Z \times X \times \bar{P}$ in $Z \times B_1 \times \bar{P} \times B_2$ ist,

so daß gilt: $\mathcal{Z}'$ und $\mathcal{Z}$ sind zueinander äquivalent.

Bew.: Da δ' als endlich vorausgesetzt wurde, läßt sich wieder die größte Länge der dritten Komponente der Argumente bestimmen; sie sei gleich r .

Es wird definiert:

$$Z := Z' \cup \{(z, x, \pi, \bar{z}, b_{4i}, p, b_j, k) / (z, x, \pi, \bar{z}, b_{4i}, p, b_j, k) \\ \in Z' \times X \times \Omega(P) \times Z' \times B_1 \times \bar{P} \times B_2 \times \{1, \dots, r+1\} \wedge |n| \leq r\}$$

$$P := P' \cup \{p_A\}, \text{ wobei } p_A \notin P'$$

δ wird folgendermaßen aus δ' konstruiert:

- 1) Gilt in $\mathcal{Z}'$ $\delta'[z, x, p] \supset \{(\bar{z}, b_{4i}, p', b_{2q})\}$ mit $p, p' \in \bar{P}'$,
so ist in $\mathcal{Z}$ aufzunehmen $\delta[z, x, p] \supset \{(\bar{z}, b_{4i}, p', b_{2q})\}$.

2) Gilt in $\mathcal{Z}'$ $\delta'[z, x, \pi] \supset \{(\bar{z}, b_{4i}, p', b_{2q})\}$, wobei $p' \in \bar{P}'$ und $\pi = p^1 \dots p^k \in \Omega(P)$ mit $1 < k \leq r$, so ist in δ aufzunehmen:

$$\begin{aligned} \delta[z, x, p^1] &\supset \{((z, x, \pi, \bar{z}, b_{4i}, p', b_{2q}, 1), b_{40}, p_A, b_{20})\} \\ (\forall j/1 \leq j \leq k) (\delta[(z, x, \pi, \bar{z}, b_{4i}, p', b_{2q}, j), x, p^j] &= \\ &= \{((z, x, \pi, \bar{z}, b_{4i}, p', b_{2q}, j+1), b_{40}, p^j, b_{24})\}) \\ (\forall p/p \in P' \setminus \{p_A\}) (\delta[(z, x, \pi, \bar{z}, b_{4i}, p', b_{2q}, k+1), x, p] &= \\ &= \{((z, x, \pi, \bar{z}, b_{4i}, p', b_{2q}, k+1), b_{40}, p, b_{24})\}) \\ \delta[(z, x, \pi, \bar{z}, b_{4i}, p', b_{2q}, k+1), x, p_A] &= \{(\bar{z}, b_{4i}, p', b_{2q})\} \end{aligned}$$

Durch vollständige Induktion über die Länge der Übergangsfolgen ergibt sich wieder:

Gilt $(z, w, \pi) \vdash_{\mathcal{Z}}^* (\bar{w}, w', \pi')$ mit $z, \bar{z} \in \mathcal{Z}'$, so gilt auch $(z, w, \pi) \vdash_{\mathcal{Z}}^* (\bar{z}, w', \pi')$ mit $z, \bar{z} \in \underline{\mathcal{Z}}'$ und umgekehrt. Daraus folgt die Behauptung unmittelbar.

Auf diese Weise braucht jeweils nur das vorderste Puffersymbol betrachtet zu werden. Diese Vorgehensweise bringt es allerdings mit sich, daß der gesamte Puffer wieder "in die richtige Reihenfolge" gebracht werden muß. Durch das Einführen des Sondersymbols p_A bleibt gewährleistet, daß durch das Rundschieben des Puffers nicht zusätzlich Ketten angenommen werden. Anzumerken ist außerdem noch, daß die Bewegung des Pufferlesekopfes (beschrieben durch b_{2q}) während der Umordnung des Puffers eleganter simuliert werden könnte; die Beschreibung würde dann jedoch umfangreicher werden.

Wegen der Transitivität der Äquivalenzbeziehung zwischen Automaten erhält man sofort die folgende Aussage:

Korollar: Zu jedem nichtdeterministischen, unvollständigen

Pufferautomaten $\mathcal{Z}'' = (X, Z, P, B_1, B_2, \delta'', z_0)$, wobei

δ'' eine endliche Korrespondenz von $Z \times X \times \Omega(P)$

in $Z \times B_1 \times \Omega(P) \times B_2$ ist, gibt es einen nichtdeterministischen, unvollständigen Pufferautomaten

$\mathcal{L}' = (X, Z', P', B_1, B_2, \delta', z_0)$, wobei δ' eine endliche Korrespondenz von $Z' \times X \times \bar{P}'$ in $Z' \times B_1 \times \bar{P}' \times B_2$ ist, so daß gilt: $\mathcal{L}^*$ und $\mathcal{L}'$ sind zueinander äquivalent.

3. Vergleich einiger Pufferautomatentypen

Im letzten Abschnitt zeigten wir die enge Verwandtschaft von Pufferautomaten und Turingmaschinen auf. So erscheint eine weitere Beschäftigung mit diesem Typ des Pufferautomaten als überflüssig.

Wir wollen hier die Allgemeinheit der Definition in verschiedener Weise einschränken und die sich so ergebenden Automatentypen miteinander vergleichen.

Wie wir sahen, beruhen die im letzten Abschnitt aufgezeigten Eigenschaften im wesentlichen auf der Fähigkeit des Automaten, sein Eingabeband "anhalten" zu können und in dieser Zeit eine beliebig große Zahl von Änderungen des Zustandes und vor allem des Puffers ausführen zu können. Wir wollen deshalb Automaten definieren, denen dies nicht erlaubt ist, die also bei jedem Zustandsübergang und jeder Pufferänderung das Eingabeband um jeweils ein Feld vorrücken müssen. Es sind dies nichtdeterministische, unvollständige Pufferautomaten, wie sie oben definiert wurden, mit der Nebenbedingung, daß B_1 nur das Element b_1 enthält. Die Pufferlesekopf-Bewegung soll ebenfalls etwas eingeschränkt werden: Wir wollen fordern, daß der Pufferlesekopf um höchstens so viele Felder in Richtung auf den Pufferschreibkopf hin bewegt werden darf, wie die Länge der maximalen Kette, die bei einem Zustandsübergang gelesen wird, beträgt. Wegen der Endlichkeit der Korrespondenz δ ist diese Länge ja wohldefiniert. Da wir Schreibarbeit sparen wollen, führen wir eine neue Definition ein:

Def.: Ein einfacher Pufferautomat ist ein Sextupel

$\mathfrak{S} = (X, Z, P, B_2, \delta, z_0)$. Dabei sind X, Z und P endliche, nichtleere Mengen, $B_2 := \{b_{2_0}, b_{2_1}, \dots, b_{2_m}\}$, wobei m gleich

der Länge der größten Kette ist, die als dritte Komponente eines Argumentes der endlichen Korrespondenz δ auftritt, $z_0 \in Z$, und δ ist eine endliche Korrespondenz von $Z \times X \times \Omega(P)$ in $Z \times \Omega(P) \times B_2$.

Anmerkung: Es soll nicht verlangt werden, daß sich der Pufferlesekopf um höchstens so viele Felder in Richtung auf den Pufferschreibkopf hin bewegt, wie die Länge der bei dem betreffenden Zustandsübergang gelesenen Kette auf dem Pufferband war. D.h., war der Übergang nur von z.B. einem Symbol abhängig, so kann sich der Pufferlesekopf dennoch um mehr als ein Symbol nach rechts bewegen, falls $m > 1$ war.

Auch für diesen Automatentyp soll die Relation $\vdash^*$ eingeführt werden, die seine Verhaltensweise beschreibt. Da sie ganz analog wie in 1. definiert wird und da durch den angefügten Buchstaben jeweils Klarheit über den betrachteten Automaten besteht, wollen wir sie gleich bezeichnen.

Def.: Seien $z, z' \in Z, x \in X, w \in \Omega(X)$, $\pi_i, \pi_k, \pi_l, \pi_r, \pi_t, \pi \in \Omega(P)$,

$$b_{2s} \in B_2.$$

Dann gelte $(z, xw, \pi_i \pi_l \pi) \vdash_{\delta}^* (z', w, \pi_l \pi \pi_r)$,

falls $(z', \pi_r, b_{2s}) \in \delta[z, x, \pi_k] \wedge \pi_i \pi_l = \pi_k \neq \wedge \wedge |\pi_i| = s$;

$$(z, xw, \pi_k \pi_t \pi) \vdash_{\delta}^* (z', w, \pi \pi_r),$$

falls $(z', \pi_r, b_{2s}) \in \delta[z, x, \pi_k] \wedge \pi_k \neq \wedge \wedge |\pi_k \pi_t| = s \neq 0$

und $(z, xw, \wedge) \vdash_{\delta}^* (z', w, \pi_r)$

falls $(z', \pi_r, b_{2s}) \in \delta[z, x, \wedge]$.

Für $x^i \in X$ ($1 \leq i \leq k$) gelte $(z, x^1 x^2 \dots x^k w, \pi) \vdash_{\delta}^* (z', w, \pi')$,

falls $z = z^1, z^2, \dots, z^{k+1} = z' \in Z, \pi = \pi^1, \dots, \pi^{k+1} = \pi' \in \Omega(P)$

existieren, so daß für $1 \leq i \leq k-1$ gilt:

$(z^i, x^1 \dots x^k w, \pi^i) \vdash_{\delta}^* (z^{i+1}, x^{i+1} \dots x^k w, \pi^{i+1})$ und außerdem

$$(z^k, x^k w, \pi^k) \vdash_{\delta}^* (z', w, \pi').$$

Def.: Sei $\mathcal{Z}$ der oben definierte einfache Pufferautomat.

Ein Wort $w \in \Omega(X)$ heißt vom einfachen Pufferautomaten $\mathcal{Z}$ angenommen, wenn gilt: Es gibt in $\mathcal{Z}$ eine Übergangsfolge

$$(z_0, w, \wedge) \vdash_{\mathcal{Z}}^* (z, \wedge, \wedge) \text{ mit } z \in Z.$$

Die Menge der vom einfachen Pufferautomaten $\mathcal{Z}$ angenommenen Wörter $w \in \Omega(X)$ werde mit $T(\mathcal{Z})$ bezeichnet.

Im folgenden Satz wird eine Aussage über den Zusammenhang zwischen einfachen Pufferautomaten, die ihr Verhalten von höchstens den beiden jeweils vordersten Symbolen auf dem Pufferband abhängig machen und solchen, die nur das jeweils vorderste Symbol berücksichtigen können, gemacht werden.

Satz 3-1: Sei $\mathcal{Z}_2 = (X, Z, P, B_2, \delta, z_0)$ ein einfacher Pufferautomat mit der endlichen Korrespondenz δ von $Z \times X \times \Omega(P)$ in $Z \times \Omega(P) \times B_2$, wobei die dritte Komponente eines beliebigen Argumentes von δ höchstens Ketten der Länge 2 enthalten darf. Dann gibt es einen einfachen Pufferautomaten $\mathcal{Z} = (X', Z', P', B'_2, \delta', z'_0)$ mit der endlichen Korrespondenz δ' von $Z' \times X' \times \bar{P}$ in $Z' \times \Omega(P') \times B'_2$, so daß gilt: $\mathcal{Z}$ ist verwandt mit $\mathcal{Z}_2$.

Bew.: Der Beweis wird konstruktiv geführt. Zunächst soll er in seinen Grundzügen skizziert werden. Um zu erreichen, daß bei jedem Zustandsübergang der Pufferautomat $\mathcal{Z}$ ebensoviel Information auf seinem Pufferband lesen kann wie der Pufferautomat $\mathcal{Z}_2$, muß die in den Puffer zu schreibende Information codiert werden. Wir werden den Pufferautomaten $\mathcal{Z}$ so konstruieren, daß jeweils analog zum Einschreiben zweier Symbole in den Puffer von $\mathcal{Z}_2$ ein codiertes Symbol in den Puffer von $\mathcal{Z}$ geschrieben wird. Dazu muß das

Pufferalphabet von $\mathcal{X}$ gegenüber dem von $\mathcal{X}_2$ erweitert werden. Diese Codierung bringt zweierlei Schwierigkeiten mit sich: 1) Soll eine Kette von ungerader Länge eingeschrieben werden, so darf das "übrigbleibende" Symbol nicht verloren gehen. 2) Soll in $\mathcal{X}_2$ der Pufferlesekopf nur um ein Feld nach rechts gehen, so darf in $\mathcal{X}$ der Pufferlesekopf nicht stehen bleiben, da sonst beim nächsten Übergang nur ein "neues" Puffersymbol zur Verfügung stünde, evtl. aber zwei gelesen werden sollen, sondern er muß ebenfalls um ein Feld nach rechts gehen; das durch die Codierung in dem ausgelesenen Symbol implizit enthaltene, noch nicht berücksichtigte Symbol, das dem zweiten Symbol in $\mathcal{X}_2$ entspricht, darf ebenfalls nicht verloren gehen. Diese Probleme werden durch eine entsprechende Erweiterung der Zustandsmenge von $\mathcal{X}$ gelöst. Da das Annehmen einer Kette von Eingabesymbolen dadurch definiert ist, daß nach dem Einlesen des letzten Symbolen der Kette der Puffer leer sein muß, müssen wir noch folgendes beachten: Der Puffer von $\mathcal{X}$ kann leer sein, obwohl eigentlich noch ein Symbol einzuschreiben wäre, das dann durch einen besonderen Zustand charakterisiert ist. Um dies nachzuprüfen, benötigen wir im Automaten $\mathcal{X}$ noch einen zusätzlichen Zustandsübergang, der durch das Anfügen eines besonderen Symbolen $\#$ an die Eingabekette von $\mathcal{X}_2$ ermöglicht wird.

Um zu zeigen, daß $\mathcal{X}$ verwandt mit $\mathcal{X}_2$ ist, werden wir jetzt folgendes zu beweisen haben:

$$u \in T(\mathcal{X}_2) \Leftrightarrow u\# \in T(\mathcal{X}), \text{ wobei } u \neq \Lambda.$$

Es genügt also, als Kette γ lediglich das Symbol $\#$ anzu-

nehmen.

Es seien: $X' := X \cup \{*\}$, $Z' := \{(P \cup \{0\}) \times Z \times (P \cup \{0\})\}$,

$P' := P \times P$, $Z_0 := (0, z_0, 0)$. Nach der Definition des einfachen Pufferautomaten gilt: $B_2 := \{b_{20}, b_{21}, b_{22}\}$ und

$B'_2 := \{b_{20}, b_{21}\}$. Die Codierung der Puffersymbole erfolgt einfach durch die entsprechende Paarbildung.

Die erste Komponente eines Elementes aus Z' gibt an, welches Puffersymbol (entschlüsselt) schon ausgelesen wurde, ohne daß das entsprechende Symbol im Pufferautomaten $\mathcal{S}_2$ ausgelesen wurde, die zweite Komponente bezeichnet den entsprechenden Zustand von $\mathcal{S}_2$, während die dritte Komponente dasjenige Symbol repräsentiert, das in $\mathcal{S}_2$ als letztes in den Puffer eingeschrieben wurde, in $\mathcal{S}$ aber wegen der ungeraden Anzahl von einzuschreibenden Symbolen noch nicht in den Puffer gelangt ist. Eine 0 in der ersten oder dritten Komponente zeigt an, daß keine Symbole gemerkt werden müssen.

Wir wollen nun die Konstruktion von δ' in Abhängigkeit von δ angeben. Für die einzelnen Typen von Übergängen in $\mathcal{S}_2$ muß ^{in $\mathcal{S}$} unterschieden werden zwischen einzuschreibenden Ketten, die eine gerade Anzahl von Symbolen umfassen und solchen mit einer ungeraden Anzahl. Die entsprechenden Relationen δ' sind aber natürlich von ähnlichem Aussehen.

1a) Gilt in $\mathcal{S}_2$ $\delta[z, x, p^1 p^2] \supset \{(z', p_1 p_2 \dots p_m, b_{22})\}$, so ist in δ' aufzunehmen:

Für gerade m:

$$\delta'[(0, z, 0), x, (p^1, p^2)] \supset \{(0, z', 0), \pi, b_{21}\} \quad 1)$$

$$\begin{aligned} \pi &:= (p_1, p_2) \dots (p_{m-1}, p_m) \\ \pi^1 &:= (p_1, p_2) \dots (p_{m-2}, p_{m-1}) \\ \pi^2 &:= (p^2, p_1) \dots (p_{m-2}, p_{m-1}) \end{aligned}$$

$$\begin{aligned} \bar{\pi} &:= (p, p_1) \dots (p_{m-2}, p_{m-1}) \\ \bar{\pi}^1 &:= (p, p_1) \dots (p_{m-1}, p_m) \\ \bar{\pi}^2 &:= (p^2, p_1) \dots (p_{m-1}, p_m) \end{aligned}$$

$$(\forall (0, z, p) / (0, z, p) \in \{0\} \times \{z\} \times P)$$

$$(\delta'[(0, z, p), x, (p^*, p^*)] \supset \{((0, z', p_m), \bar{\pi}, b_{z_1})\})$$

$$(\forall (p^2, p^i) / (p^2, p^i) \in \{p^2\} \times P)$$

$$(\delta'[(p^*, z, 0), x, (p^2, p^i)] \supset \{((p^i, z', 0), \pi, b_{z_1})\})$$

$$(\forall (p^2, p^i) / (p^2, p^i) \in \{p^2\} \times P) (\forall (p^*, z, p) / (p^*, z, p) \in \{p^*\} \times \{z\} \times P)$$

$$(\delta'[(p^*, z, p), x, (p^2, p^i)] \supset \{((p^i, z', p_m), \bar{\pi}, b_{z_1})\})$$

$$\delta'[(p^*, z, p^2), x, \wedge] \supset \{((0, z', 0), \pi, b_{z_0})\}$$

Für ungerade m:

$$\delta'[(0, z, 0), x, (p^*, p^2)] \supset \{((0, z', p_m), \pi', b_{z_1})\}$$

$$(\forall (0, z, p) / (0, z, p) \in \{0\} \times \{z\} \times P)$$

$$(\delta'[(0, z, p), x, (p^*, p^2)] \supset \{((0, z', 0), \bar{\pi}', b_{z_1})\})$$

$$(\forall (p^2, p^i) / (p^2, p^i) \in \{p^2\} \times P)$$

$$(\delta'[(p^*, z, 0), x, (p^2, p^i)] \supset \{((p^i, z', p_m), \pi', b_{z_1})\})$$

$$(\forall (p^2, p^i) / (p^2, p^i) \in \{p^2\} \times P) (\forall (p^*, z, p) / (p^*, z, p) \in \{p^*\} \times \{z\} \times P)$$

$$(\delta'[(p^*, z, p), x, (p^2, p^i)] \supset \{((p^i, z', 0), \bar{\pi}', b_{z_1})\})$$

$$\delta'[(p^*, z, p^2), x, \wedge] \supset \{((0, z', p_m), \pi', b_{z_0})\}$$

1b) Gilt in $\mathcal{S}_2$ $\delta[z, x, p^* p^2] \supset \{(z', \wedge, b_{z_2})\}$, so ist in δ' aufzunehmen:

$$\delta'[(0, z, 0), x, (p^*, p^2)] \supset \{((0, z', 0), \wedge, b_{z_1})\}$$

$$(\forall (0, z, p) / (0, z, p) \in \{0\} \times \{z\} \times P)$$

$$(\delta'[(0, z, p), x, (p^*, p^2)] \supset \{((0, z', p), \wedge, b_{z_1})\})$$

$$(\forall (p^2, p^i) / (p^2, p^i) \in \{p^2\} \times P)$$

$$(\delta'[(p^*, z, 0), x, (p^2, p^i)] \supset \{((p^i, z', 0), \wedge, b_{z_1})\})$$

$$(\forall (p^*, z, p) / (p^*, z, p) \in \{p^*\} \times \{z\} \times P) (\forall (p^2, p^i) / (p^2, p^i) \in \{p^2\} \times P)$$

$$(\delta'[(p^*, z, p), x, (p^2, p^i)] \supset \{((p^i, z', p), \wedge, b_{z_1})\})$$

$$\delta'[(p^*, z, p^2), x, \wedge] \supset \{((0, z', 0), \wedge, b_{z_0})\}$$

2a) Gilt in $\mathcal{S}_2$ $\delta[z, x, p^* p^2] \supset \{(z', p_1 p_2 \dots p_m, b_{z_1})\}$, so ist in δ' aufzunehmen:

Für gerade m :

$$\begin{aligned} \delta'[(0, z, 0), x, (p^1, p^2)] &\supset \{((p^2, z', 0), \pi, b_{24})\} \\ (\forall (0, z, p)/(0, z, p) \in \{0\} \times \{z\} \times P) \\ &(\delta'[(0, z, p), x, (p^1, p^2)] \supset \{((p^2, z', p_m), \bar{\pi}, b_{24})\}) \\ (\forall (p^1, p^2)/(p^1, p^2) \in \{p^1\} \times P) \\ &(\delta'[(p^1, z, 0), x, (p^2, p^2)] \supset \{((0, z', 0), \pi, b_{20})\}) \\ (\forall (p^1, z, p)/(p^1, z, p) \in \{p^1\} \times \{z\} \times P) (\forall (p^2, p^2)/(p^2, p^2) \in \{p^2\} \times P) \\ &(\delta'[(p^1, z, p), x, (p^2, p^2)] \supset \{((0, z', p_m), \bar{\pi}, b_{20})\}) \\ \delta'[(p^1, z, p^2), x, \wedge] &\supset \{((0, z', p_m), \bar{\pi}, b_{20})\} \end{aligned}$$

Für ungerade m :

$$\begin{aligned} \delta'[(0, z, 0), x, (p^1, p^2)] &\supset \{((p^2, z', p_m), \pi', b_{24})\} \\ (\forall (0, z, p)/(0, z, p) \in \{0\} \times \{z\} \times P) \\ &(\delta'[(0, z, p), x, (p^1, p^2)] \supset \{((p^2, z', 0), \bar{\pi}', b_{24})\}) \\ (\forall (p^1, p^2)/(p^1, p^2) \in \{p^1\} \times P) \\ &(\delta'[(p^1, z, 0), x, (p^2, p^2)] \supset \{((0, z', p_m), \pi', b_{20})\}) \\ (\forall (p^1, z, p)/(p^1, z, p) \in \{p^1\} \times \{z\} \times P) (\forall (p^2, p^2)/(p^2, p^2) \in \{p^2\} \times P) \\ &(\delta'[(p^1, z, p), x, (p^2, p^2)] \supset \{((0, z', 0), \bar{\pi}', b_{20})\}) \\ \delta'[(p^1, z, p^2), x, \wedge] &\supset \{((0, z', 0), \bar{\pi}', b_{20})\} \end{aligned}$$

2b) Gilt in $\mathcal{L}_2$ $\delta[z, x, p^1 p^2] \supset \{(z', \wedge, b_{24})\}$, so ist in δ' aufzunehmen:

$$\begin{aligned} \delta'[(0, z, 0), x, (p^1, p^2)] &\supset \{((p^2, z', 0), \wedge, b_{24})\} \\ (\forall (0, z, p)/(0, z, p) \in \{0\} \times \{z\} \times P) \\ &(\delta'[(0, z, p), x, (p^1, p^2)] \supset \{((p^2, z', p), \wedge, b_{24})\}) \\ (\forall (p^1, p^2)/(p^1, p^2) \in \{p^1\} \times P) \\ &(\delta'[(p^1, z, 0), x, (p^2, p^2)] \supset \{((0, z', 0), \wedge, b_{20})\}) \\ (\forall (p^1, z, p)/(p^1, z, p) \in \{p^1\} \times \{z\} \times P) (\forall (p^2, p^2)/(p^2, p^2) \in \{p^2\} \times P) \\ &(\delta'[(p^1, z, p), x, (p^2, p^2)] \supset \{((0, z', p), \wedge, b_{20})\}) \\ \delta'[(p^1, z, p^2), x, \wedge] &\supset \{((0, z', p^2), \wedge, b_{20})\} \end{aligned}$$

3a) Gilt in $\mathcal{L}_1$ $\delta[z, x, p^* p^2] \supset \{(z', p_1 \dots p_m, b_{20})\}$, so ist in δ' aufzunehmen:

Für gerade m:

$$\begin{aligned} \delta'[(0, z, 0), x, (p^*, p^2)] &\supset \{((0, z', 0), \pi, b_{20})\} \\ (\forall (0, z, p)/(0, z, p) \in \{0\} \times \{z\} \times P) \\ &(\delta'[(0, z, p), x, (p^*, p^2)] \supset \{((0, z', p_m), \bar{\pi}, b_{20})\}) \\ (\forall (p^2, p^i)/(p^2, p^i) \in \{p^2\} \times P) \\ &(\delta'[(p^*, z, 0), x, (p^2, p^i)] \supset \{((p^*, z', 0), \pi, b_{20})\}) \\ (\forall (p^*, z, p)/(p^*, z, p) \in \{p^*\} \times \{z\} \times P) (\forall (p^2, p^i)/(p^2, p^i) \in \{p^2\} \times P) \\ &(\delta'[(p^*, z, p), x, (p^2, p^i)] \supset \{((p^*, z', p_m), \bar{\pi}, b_{20})\}) \\ \delta'[(p^*, z, p^2), x, \wedge] &\supset \{((p^*, z', p_m), \bar{\pi}^*, b_{20})\} \end{aligned}$$

Für ungerade m:

$$\begin{aligned} \delta'[(0, z, 0), x, (p^*, p^2)] &\supset \{((0, z', p_m), \pi, b_{20})\} \\ (\forall (0, z, p)/(0, z, p) \in \{0\} \times \{z\} \times P) \\ &(\delta'[(0, z, p), x, (p^*, p^2)] \supset \{((0, z', 0), \bar{\pi}', b_{20})\}) \\ (\forall (p^2, p^i)/(p^2, p^i) \in \{p^2\} \times P) \\ &(\delta'[(p^*, z, 0), x, (p^2, p^i)] \supset \{((p^*, z', p_m), \pi', b_{20})\}) \\ (\forall (p^*, z, p)/(p^*, z, p) \in \{p^*\} \times \{z\} \times P) (\forall (p^2, p^i)/(p^2, p^i) \in \{p^2\} \times P) \\ &(\delta'[(p^*, z, p), x, (p^2, p^i)] \supset \{((p^*, z', 0), \bar{\pi}', b_{20})\}) \\ \delta'[(p^*, z, p^2), x, \wedge] &\supset \{((p^*, z', 0), \bar{\pi}^*, b_{20})\} \end{aligned}$$

3b) Gilt in $\mathcal{L}_1$ $\delta[z, x, p^* p^2] \supset \{(z', \wedge, b_{20})\}$, so ist in δ' aufzunehmen:

$$\begin{aligned} \delta'[(0, z, 0), x, (p^*, p^2)] &\supset \{((0, z', 0), \wedge, b_{20})\} \\ (\forall (0, z, p)/(0, z, p) \in \{0\} \times \{z\} \times P) \\ &(\delta'[(0, z, p), x, (p^*, p^2)] \supset \{((0, z', p), \wedge, b_{20})\}) \\ (\forall (p^2, p^i)/(p^2, p^i) \in \{p^2\} \times P) \\ &(\delta'[(p^*, z, 0), x, (p^2, p^i)] \supset \{((p^*, z', 0), \wedge, b_{20})\}) \\ (\forall (p^*, z, p)/(p^*, z, p) \in \{p^*\} \times \{z\} \times P) (\forall (p^2, p^i)/(p^2, p^i) \in \{p^2\} \times P) \\ &(\delta'[(p^*, z, p), x, (p^2, p^i)] \supset \{((p^*, z', p), \wedge, b_{20})\}) \end{aligned}$$

$$\delta'[(p^*, z, p^*), x, \wedge] \supset \{((p^*, z', p^*), \wedge, b_{20})\}$$

4a) Gilt in $\mathcal{S}_2$ $\delta'[(z, x, p^*)] \supset \{(z', p_m \dots p_m, b_{22})\}$, so ist in δ' aufzunehmen:

Für gerade m:

$$\begin{aligned} & (\forall (p^*, p^i)/(p^*, p^i) \in \{p^i\} \times P) \\ & \quad (\delta'[(0, z, 0), x, (p^*, p^i)] \supset \{((0, z', 0), \pi, b_{24})\}) \\ & (\forall (0, z, p)/(0, z, p) \in \{0\} \times \{z\} \times P) (\forall (p^*, p^i)/(p^*, p^i) \in \{p^i\} \times P) \\ & \quad (\delta'[(0, z, p), x, (p^*, p^i)] \supset \{((0, z', p_m), \bar{\pi}, b_{24})\}) \\ & (\forall (p^i, p^k)/(p^i, p^k) \in P') \\ & \quad (\delta'[(p^*, z, 0), x, (p^i, p^k)] \supset \{((p^k, z', 0), \pi, b_{24})\}) \\ & (\forall (p^*, z, p)/(p^*, z, p) \in \{p^i\} \times \{z\} \times P) (\forall (p^i, p^k)/(p^i, p^k) \in P') \\ & \quad (\delta'[(p^*, z, p), x, (p^i, p^k)] \supset \{((p^k, z', p_m), \bar{\pi}, b_{24})\}) \\ & \delta'[(p^*, z, 0), x, \wedge] \supset \{((0, z', 0), \pi, b_{20})\} \\ & (\forall (p^*, z, p)/(p^*, z, p) \in \{p^i\} \times \{z\} \times P) \\ & \quad (\delta'[(p^*, z, p), x, \wedge] \supset \{((0, z', 0), \pi, b_{20})\}) \\ & \delta'[(0, z, p^*), x, \wedge] \supset \{((0, z', 0), \pi, b_{20})\} \end{aligned}$$

Für ungerade m:

$$\begin{aligned} & (\forall (p^*, p^i)/(p^*, p^i) \in \{p^i\} \times P) \\ & \quad (\delta'[(0, z, 0), x, (p^*, p^i)] \supset \{((0, z', p_m), \pi', b_{24})\}) \\ & (\forall (0, z, p)/(0, z, p) \in \{0\} \times \{z\} \times P) (\forall (p^*, p^i)/(p^*, p^i) \in \{p^i\} \times P) \\ & \quad (\delta'[(0, z, p), x, (p^*, p^i)] \supset \{((0, z', 0), \bar{\pi}', b_{24})\}) \\ & (\forall (p^i, p^k)/(p^i, p^k) \in P') \\ & \quad (\delta'[(p^*, z, 0), x, (p^i, p^k)] \supset \{((p^k, z', p_m), \pi', b_{24})\}) \\ & (\forall (p^*, z, p)/(p^*, z, p) \in \{p^i\} \times \{z\} \times P) (\forall (p^i, p^k)/(p^i, p^k) \in P') \\ & \quad (\delta'[(p^*, z, p), x, (p^i, p^k)] \supset \{((p^k, z', 0), \bar{\pi}', b_{24})\}) \\ & \delta'[(p^*, z, 0), x, \wedge] \supset \{((0, z', p_m), \pi', b_{20})\} \\ & (\forall (p^*, z, p)/(p^*, z, p) \in \{p^i\} \times \{z\} \times P) \\ & \quad (\delta'[(p^*, z, p), x, \wedge] \supset \{((0, z', p_m), \pi', b_{20})\}) \\ & \delta'[(0, z, p^*), x, \wedge] \supset \{((0, z', p_m), \pi', b_{20})\} \end{aligned}$$

4b) Gilt in $\mathfrak{S}_1$ $\delta[z, x, p'] \supset \{(z', \wedge, b_{22})\}$, so ist in δ' aufzunehmen:

$$\begin{aligned}
 & (\forall (p^a, p^i) / (p^a, p^i) \in \{p^a\} \times P) \\
 & \quad (\delta'[(0, z, 0), x, (p^a, p^i)] \supset \{((0, z', 0), \wedge, b_{21})\}) \\
 & (\forall (0, z, p) / (0, z, p) \in \{0\} \times \{z\} \times P) (\forall (p^a, p^i) / (p^a, p^i) \in \{p^a\} \times P) \\
 & \quad (\delta'[(0, z, p), x, (p^a, p^i)] \supset \{((0, z', p), \wedge, b_{21})\}) \\
 & (\forall (p^i, p^k) / (p^i, p^k) \in P') \\
 & \quad (\delta'[(p^a, z, 0), x, (p^i, p^k)] \supset \{((p^k, z', 0), \wedge, b_{21})\}) \\
 & (\forall (p^a, z, p) / (p^a, z, p) \in \{p^a\} \times \{z\} \times P) (\forall (p^i, p^k) / (p^i, p^k) \in P') \\
 & \quad (\delta'[(p^a, z, p), x, (p^i, p^k)] \supset \{((p^k, z', p), \wedge, b_{21})\}) \\
 & \quad \delta'[(p^a, z, 0), x, \wedge] \supset \{((0, z', 0), \wedge, b_{20})\} \\
 & (\forall (p^a, z, p) / (p^a, z, p) \in \{p^a\} \times \{z\} \times P) \\
 & \quad (\delta'[(p^a, z, p), x, \wedge] \supset \{((0, z', 0), \wedge, b_{20})\}) \\
 & \quad \delta'[(0, z, p^a), x, \wedge] \supset \{((0, z', 0), \wedge, b_{20})\}
 \end{aligned}$$

5a) Gilt in $\mathfrak{S}_1$ $\delta[z, x, p'] \supset \{(z', p_1 \dots p_m, b_{21})\}$, so ist in δ' aufzunehmen:

Für gerade m:

$$\begin{aligned}
 & (\forall (p^a, p^i) / (p^a, p^i) \in \{p^a\} \times P) \\
 & \quad (\delta'[(0, z, 0), x, (p^a, p^i)] \supset \{((p^i, z', 0), \pi, b_{21})\}) \\
 & (\forall (0, z, p) / (0, z, p) \in \{0\} \times \{z\} \times P) (\forall (p^a, p^i) / (p^a, p^i) \in \{p^a\} \times P) \\
 & \quad (\delta'[(0, z, p), x, (p^a, p^i)] \supset \{((p^i, z', p_m), \bar{\pi}, b_{21})\}) \\
 & (\forall (p^i, p^k) / (p^i, p^k) \in P') \\
 & \quad (\delta'[(p^a, z, 0), x, (p^i, p^k)] \supset \{((0, z', 0), \pi, b_{20})\}) \\
 & (\forall (p^a, z, p) / (p^a, z, p) \in \{p^a\} \times \{z\} \times P) (\forall (p^i, p^k) / (p^i, p^k) \in P') \\
 & \quad (\delta'[(p^a, z, p), x, (p^i, p^k)] \supset \{((0, z', p_m), \bar{\pi}, b_{20})\}) \\
 & \quad \delta'[(p^a, z, 0), x, \wedge] \supset \{((0, z', 0), \pi, b_{20})\} \\
 & (\forall (p^a, z, p) / (p^a, z, p) \in \{p^a\} \times \{z\} \times P) \\
 & \quad (\delta'[(p^a, z, p), x, \wedge] \supset \{((0, z', p_m), \bar{\pi}, b_{20})\}) \\
 & \quad \delta'[(0, z, p^a), x, \wedge] \supset \{((0, z', 0), \pi, b_{20})\}
 \end{aligned}$$

Für ungerade m :

$$\begin{aligned}
 & (\forall (p^i, p^i)/(p^i, p^i) \in \{p^i\} \times P) \\
 & \quad (\delta'[(0, z, 0), x, (p^i, p^i)] \supset \{((p^i, z', p_m), \pi', b_{2i})\}) \\
 & (\forall (0, z, p)/(0, z, p) \in \{0\} \times \{z\} \times P) (\forall (p^i, p^i)/(p^i, p^i) \in \{p^i\} \times P) \\
 & \quad (\delta'[(0, z, p), x, (p^i, p^i)] \supset \{((p^i, z', 0), \bar{\pi}', b_{2i})\}) \\
 & (\forall (p^i, p^k)/(p^i, p^k) \in P') \\
 & \quad (\delta'[(p^i, z, 0), x, (p^i, p^k)] \supset \{((0, z', p_m), \pi', b_{20})\}) \\
 & (\forall (p^i, z, p)/(p^i, z, p) \in \{p^i\} \times \{z\} \times P) (\forall (p^i, p^k)/(p^i, p^k) \in P') \\
 & \quad (\delta'[(p^i, z, p), x, (p^i, p^k)] \supset \{((0, z', 0), \bar{\pi}', b_{20})\}) \\
 & \quad \delta'[(p^i, z, 0), x, \wedge] \supset \{((0, z', p_m), \pi', b_{20})\} \\
 & (\forall (p^i, z, p)/(p^i, z, p) \in \{p^i\} \times \{z\} \times P) \\
 & \quad (\delta'[(p^i, z, p), x, \wedge] \supset \{((0, z', 0), \bar{\pi}', b_{20})\}) \\
 & \quad \delta'[(0, z, p^i), x, \wedge] \supset \{((0, z', p_m), \pi', b_{20})\}
 \end{aligned}$$

5b) Gilt in $\mathfrak{Z}_2$ $\delta[z, x, p^i] \supset \{(z', \wedge, b_{2i})\}$, so ist in δ' aufzunehmen:

$$\begin{aligned}
 & (\forall (p^i, p^i)/(p^i, p^i) \in \{p^i\} \times P) \\
 & \quad (\delta'[(0, z, 0), x, (p^i, p^i)] \supset \{((p^i, z', 0), \wedge, b_{2i})\}) \\
 & (\forall (0, z, p)/(0, z, p) \in \{0\} \times \{z\} \times P) (\forall (p^i, p^i)/(p^i, p^i) \in \{p^i\} \times P) \\
 & \quad (\delta'[(0, z, p), x, (p^i, p^i)] \supset \{((p^i, z', p), \wedge, b_{2i})\}) \\
 & (\forall (p^i, p^k)/(p^i, p^k) \in P') \\
 & \quad (\delta'[(p^i, z, 0), x, (p^i, p^k)] \supset \{((0, z', 0), \wedge, b_{20})\}) \\
 & (\forall (p^i, z, p)/(p^i, z, p) \in \{p^i\} \times \{z\} \times P) (\forall (p^i, p^k)/(p^i, p^k) \in P') \\
 & \quad (\delta'[(p^i, z, p), x, (p^i, p^k)] \supset \{((0, z', p), \wedge, b_{20})\}) \\
 & \quad \delta'[(p^i, z, 0), x, \wedge] \supset \{((0, z', 0), \wedge, b_{20})\} \\
 & (\forall (p^i, z, p)/(p^i, z, p) \in \{p^i\} \times \{z\} \times P) \\
 & \quad (\delta'[(p^i, z, p), x, \wedge] \supset \{((0, z', p), \wedge, b_{20})\}) \\
 & \quad \delta'[(0, z, p^i), x, \wedge] \supset \{((0, z', 0), \wedge, b_{20})\}
 \end{aligned}$$

6a) Gilt in $\mathfrak{Z}_2$ $\delta[z, x, p^i] \supset \{(z', p_1 \dots p_m, b_{20})\}$, so ist in δ' aufzunehmen:

Für gerade m:

$$\begin{aligned}
 & (\forall (p^1, p^i) / (p^1, p^i) \in \{p^1\} \times P) \\
 & \quad (\delta'[(0, z, 0), x, (p^1, p^i)] \supset \{((0, z', 0), \pi, b_{20})\}) \\
 & (\forall (0, z, p) / (0, z, p) \in \{0\} \times \{z\} \times P) (\forall (p^1, p^i) / (p^1, p^i) \in \{p^1\} \times P) \\
 & \quad (\delta'[(0, z, p), x, (p^1, p^i)] \supset \{(0, z', p_m), \bar{\pi}, b_{20}\}) \\
 & (\forall (p^i, p^k) / (p^i, p^k) \in P^i) \\
 & \quad (\delta'[(p^1, z, 0), x, (p^i, p^k)] \supset \{((p^1, z', 0), \pi, b_{20})\}) \\
 & (\forall (p^1, z, p) / (p^1, z, p) \in \{p^1\} \times \{z\} \times P) (\forall (p^i, p^k) / (p^i, p^k) \in P^i) \\
 & \quad (\delta'[(p^1, z, p), x, (p^i, p^k)] \supset \{((p^1, z', p_m), \bar{\pi}, b_{20})\}) \\
 & \delta'[(p^1, z, 0), x, \wedge] \supset \{((p^1, z', 0), \pi, b_{20})\} \\
 & (\forall (p^1, z, p) / (p^1, z, p) \in \{p^1\} \times \{z\} \times P) \\
 & \quad (\delta'[(p^1, z, p), x, \wedge] \supset \{((p^1, z', p_m), \bar{\pi}, b_{20})\}) \\
 & \delta'[(0, z, p^1), x, \wedge] \supset \{((0, z', p_m), (p^1, p_1) \dots (p_{m-2}, p_{m-1}), b_{20})\}
 \end{aligned}$$

Für ungerade m:

$$\begin{aligned}
 & (\forall (p^1, p^i) / (p^1, p^i) \in \{p^1\} \times P) \\
 & \quad (\delta'[(0, z, 0), x, (p^1, p^i)] \supset \{((0, z', p_m), \pi', b_{20})\}) \\
 & (\forall (0, z, p) / (0, z, p) \in \{0\} \times \{z\} \times P) (\forall (p^1, p^i) / (p^1, p^i) \in \{p^1\} \times P) \\
 & \quad (\delta'[(0, z, p), x, (p^1, p^i)] \supset \{((0, z', 0), \bar{\pi}', b_{20})\}) \\
 & (\forall (p^i, p^k) / (p^i, p^k) \in P^i) \\
 & \quad (\delta'[(p^1, z, 0), x, (p^i, p^k)] \supset \{((p^1, z', p_m), \pi', b_{20})\}) \\
 & (\forall (p^1, z, p) / (p^1, z, p) \in \{p^1\} \times \{z\} \times P) (\forall (p^i, p^k) / (p^i, p^k) \in P^i) \\
 & \quad (\delta'[(p^1, z, p), x, (p^i, p^k)] \supset \{((p^1, z', 0), \bar{\pi}', b_{20})\}) \\
 & \delta'[(p^1, z, 0), x, \wedge] \supset \{((p^1, z', p_m), \pi', b_{20})\} \\
 & (\forall (p^1, z, p) / (p^1, z, p) \in \{p^1\} \times \{z\} \times P) \\
 & \quad (\delta'[(p^1, z, p), x, \wedge] \supset \{((p^1, z', 0), \bar{\pi}', b_{20})\}) \\
 & \delta'[(0, z, p^1), x, \wedge] \supset \{((0, z', 0), (p^1, p_1) \dots (p_{m-1}, p_m), b_{20})\}
 \end{aligned}$$

6b) Gilt in $\mathfrak{Z}_2$ $\delta[z, x, p^1] \supset \{(z', \wedge, b_{20})\}$, so ist in δ' aufzunehmen:

$$\begin{aligned}
 & (\forall (p^1, p^i) / (p^1, p^i) \in \{p^1\} \times P) \\
 & \quad (\delta'[(0, z, 0), x, (p^1, p^i)] \supset \{((0, z', 0), \wedge, b_{20})\})
 \end{aligned}$$

$$(\forall (0, z, p) / (0, z, p) \in \{0\} \times \{z\} \times P) (\forall (p^i, p^k) / (p^i, p^k) \in \{p^i\} \times P) \\ (\delta'[(0, z, p), x, (p^i, p^k)] \supset \{((0, z', p), \wedge, b_{20})\})$$

$$(\forall (p^i, p^k) / (p^i, p^k) \in P')$$

$$(\delta'[(p^i, z, 0), x, (p^i, p^k)] \supset \{((p^i, z', 0), \wedge, b_{20})\})$$

$$(\forall (p^i, z, p) / (p^i, z, p) \in \{p^i\} \times \{z\} \times P) (\forall (p^i, p^k) / (p^i, p^k) \in P')$$

$$(\delta'[(p^i, z, p), x, (p^i, p^k)] \supset \{((p^i, z', p), \wedge, b_{20})\})$$

$$\delta'[(0, z, p^i), x, \wedge] \supset \{((0, z', p^i), \wedge, b_{20})\}$$

7) Gilt in $\mathfrak{S}_2$ $\delta[z, x, \wedge] \supset \{(z', p_1 \dots p_m, b_{2i})$ mit $i \in \{0, 1, 2\}$,
so ist in δ' aufzunehmen:

Für gerade m:

$$\delta'[(0, z, 0), x, \wedge] \supset \{((0, z', 0), \pi, b_{20})\}$$

Für ungerade m:

$$\delta'[(0, z, 0), x, \wedge] \supset \{((0, z', p_m), \pi', b_{20})\}$$

Gilt in $\mathfrak{S}_1$ $\delta[z, x, \wedge] \supset \{(z', \wedge, b_{2i})\}$ mit $i \in \{0, 1, 2\}$,

so ist in δ' aufzunehmen:

$$\delta'[(0, z, 0), x, \wedge] \supset \{((0, z', 0), \wedge, b_{20})\}$$

Außerdem muß in $\mathfrak{S}$ noch die Behandlung des Symbols " $\#$ "
gewährleistet werden. Es sind dazu in δ' aufzunehmen:

$$8) (\forall (p', z, p'') / (p', z, p'') \in Z') (\forall (p^i, p^k) / (p^i, p^k) \in P') \\ (\delta'[(p', z, p''), \#, (p^i, p^k)] \supset \{((p', z', p''), (p^i, p^k), b_{20})\})$$

$$(\forall (p, z, 0) / (p, z, 0) \in P \times Z \times \{0\})$$

$$(\delta'[(p, z, 0), \#, \wedge] \supset \{((p, z, 0), (p, p), b_{20})\})$$

$$(\forall (0, z, p) / (0, z, p) \in \{0\} \times Z \times P)$$

$$(\delta'[(0, z, p), \#, \wedge] \supset \{((0, z, p), (p, p), b_{20})\})$$

$$(\forall (0, z, 0) / (0, z, 0) \in \{0\} \times Z \times \{0\})$$

$$(\delta'[(0, z, 0), \#, \wedge] \supset \{((0, z, 0), \wedge, b_{20})\})$$

Aufgrund der Konstruktion von δ' ergibt sich durch vollständige Induktion über die Länge der Übergangsfolgen:

Gibt es in $\mathfrak{S}_1$ eine Übergangsfolge, die $(z_0, w, \wedge)$ in

$(z', w', \wedge)$ überführt, wobei $w \in \Omega(X) \setminus \{\wedge\}$ und $|w| > |w'|$, so gibt es in $\mathcal{S}$ eine Übergangsfolge, die $((0, z_0, 0), w, \wedge)$ in $((0, z', 0), w', \wedge)$ überführt und umgekehrt.

Gilt dann $(z_0, w, \wedge) \vdash_{\mathcal{S}}^* (z, \wedge, \wedge)$, so gilt in $\mathcal{S}$ $((0, z_0, 0), w, \wedge) \vdash_{\mathcal{S}}^* ((0, z, 0), \wedge, \wedge)$ und umgekehrt.

Mit 3) folgt daraus:

$$w \in T(\mathcal{S}_1) \Leftrightarrow w \# \in T(\mathcal{S})$$

Damit ist gezeigt, daß $\mathcal{S}$ mit $\mathcal{S}_1$ verwandt ist, und die Behauptung ist bewiesen.

Das Annehmen von Wörtern durch einen einfachen Pufferautomaten war durch die Forderung definiert, daß vom Ausgangszustand ausgehend eine Übergangsfolge der Länge n existieren muß, -wenn n die Länge des Wortes war-, deren letzte Konfiguration durch das Leersein des Pufferbandes gekennzeichnet ist. Für den Endzustand sind keine Einschränkungen gemacht. Analog zu der bei der Behandlung von Kellerautomaten auftretenden Definition wollen wir einen Pufferautomatentyp definieren, bei dem eine Teilmenge der Zustände ausgezeichnet ist.

Def.: Ein markierter einfacher Pufferautomat ist ein Septupel

$\mathcal{S}_F = (X, Z, P, B_2, \delta, z_0, F)$. Dabei sind X, Z und P endliche, nichtleere Mengen, $B_2 = \{b_{z_0}, b_{z_1}, \dots, b_{z_m}\}$, wobei m gleich der Länge der größten Kette ist, die als dritte Komponente eines Argumentes der endlichen Korrespondenz δ auftritt, $z_0 \in Z$ und $F \subset Z$. δ ist eine endliche Korrespondenz von $Z \times X \times \Omega(P)$ in $Z \times \Omega(P) \times B_2$.

Es ist zu beachten, daß die Menge der markierten Zustände, wie F genannt wird, eine echte Teilmenge von Z darstellen muß.

Die Relation $\vdash^*$ ist wie beim einfachen Pufferautomaten definiert.

Def.: Sei $\mathcal{S}_F$ der oben definierte markierte einfache Pufferautomat.

Ein Wort $w \in \Omega(X)$ heißt vom markierten einfachen Pufferautomaten $\mathcal{S}_F$ angenommen, wenn gilt: Es gibt eine Übergangsfolge $(z_0, w, \wedge) \vdash_{\mathcal{S}_F}^* (z, \wedge, \wedge)$ mit $z \in F$.

Die Menge der von $\mathcal{S}_F$ angenommenen Wörter $w \in \Omega(X)$ werde mit $T(\mathcal{S}_F)$ bezeichnet

Über den Zusammenhang zwischen markierten einfachen Pufferautomaten und einfachen Pufferautomaten gibt der folgende Satz Auskunft:

Satz 3-2: Sei $\mathcal{S}_F = (X, Z, P, B_2, \delta, z_0, F)$ ein markierter einfacher Pufferautomat. Dann gibt es einen einfachen Pufferautomaten $\mathcal{S} = (X', Z, P, B_2, \delta', z_0)$, so daß gilt:
 $\mathcal{S}$ ist verwandt mit $\mathcal{S}_F$.

Bew.: Es wird gesetzt: $X' := X \cup \{\#\}$.

δ' wird folgendermaßen konstruiert:

$$(\forall (z, x, \pi) / (z, x, \pi) \in Z \times X \times \Omega(P))$$

$$(\delta'[z, x, \pi] := \delta[z, x, \pi])$$

$$(\forall z/z \in F) (\delta'[z, \#, \wedge] = \{(z, \wedge, b_{20})\})$$

$$(\forall z/z \in F) (\forall \pi/\pi \in P)$$

$$(\delta'[z, \#, \pi] = \{(z, \wedge, b_{20})\})$$

$$(\forall z/z \in Z \setminus F) (\forall \bar{\pi}/\bar{\pi} \in \bar{\Omega}(P)) (\delta'[z, \#, \bar{\pi}] = \{(z, p, b_{20})\})$$

mit $p \in P$.

Damit ergibt sich, daß beim letzten direkten Übergang, bei dem das Symbol $\#$ abgearbeitet wird, das Pufferband nur dann leer bleibt, -und damit das erweiterte Wort von $\mathcal{S}$ angenommen wird-, wenn es nach dem Einlesen des ursprünglichen Wortes leer war und gleichzeitig ein Zustand aus

der Menge der markierten Zustände F erreicht war.

Aus dieser Konstruktion von $\mathcal{Z}$ folgt unmittelbar, daß für $w \in \Omega(X) \setminus \{\Lambda\}$ gilt:

$$w \in T(\mathcal{Z}_F) \Leftrightarrow w\# \in T(\mathcal{Z}),$$

d.h. $\mathcal{Z}$ ist verwandt mit $\mathcal{Z}_F$.

Andererseits gilt auch

Satz 3-3: Sei $\mathcal{Z} = (X, Z, P, B_2, \delta, z_0)$ ein einfacher Pufferautomat.

Dann gibt es einen markierten einfachen Pufferautomaten $\mathcal{Z}_F = (X_F, Z, P, B_2, \delta_F, z_0, F)$, so daß gilt:

$\mathcal{Z}_F$ ist verwandt mit $\mathcal{Z}$.

Bew.: Es wird gebildet: $X_F := X \cup \{\#\}$; $F := \{z_0\}$.

δ_F wird folgendermaßen aus δ konstruiert:

$$(\forall (z, x, \pi) / (z, x, \pi) \in Z \times X \times \Omega(P)) (\delta_F[z, x, \pi] := \delta[z, x, \pi])$$

$$(\forall z / z \in Z) (\forall \pi / \pi \in P) (\delta_F[z, \#, \pi] = \{(z, \Lambda, b_{z_0})\})$$

$$(\forall z / z \in Z) (\delta_F[z, \#, \Lambda] = \{(z_0, \Lambda, b_{z_0})\})$$

Aus dieser Konstruktion folgt unmittelbar, daß für $w \in \Omega(X) \setminus \{\Lambda\}$ gilt:

$$w \in T(\mathcal{Z}) \Leftrightarrow w\# \in T(\mathcal{Z}_F);$$

d.h. $\mathcal{Z}_F$ ist verwandt mit $\mathcal{Z}$.

Bei den beiden letzten Sätzen ist es von Bedeutung, daß jeweils die Zustandsmengen von $\mathcal{Z}$ und $\mathcal{Z}_F$ gleich sind. Es gibt nämlich trivialerweise zu jedem einfachen Pufferautomaten $\mathcal{Z}$ einen einfachen markierten Pufferautomaten $\mathcal{Z}'_F$, so daß sogar gilt:

$\mathcal{Z}$ und $\mathcal{Z}'_F$ sind zueinander äquivalent. Ist nämlich $\mathcal{Z}$ gegeben durch $(X, Z, P, B_2, \delta, z_0)$ und $\mathcal{Z}'_F$ durch $(X, Z', P, B_2, \bar{\delta}, z_0, F)$, wobei $Z' := Z \cup \{\bar{z}\}$ mit $\bar{z} \notin Z$, $F = Z$ und $\bar{\delta}$ definiert ist durch

$$(\forall (z, x, \pi) / (z, x, \pi) \in Z \times X \times \Omega(P)) (\bar{\delta}[z, x, \pi] = \delta[z, x, \pi]) \text{ und}$$

$$(\forall (\bar{z}, x, \pi) / (\bar{z}, x, \pi) \in \{\bar{z}\} \times X \times \bar{P}) (\bar{\delta}[\bar{z}, x, \pi] = \emptyset), \text{ so gilt}$$

offensichtlich für $w \in \Omega(X)$: $w \in T(\mathcal{Z}) \Leftrightarrow w \in T(\mathcal{Z}'_F)$.

Die Zustandsmengen Z und Z' sind hierbei aber nicht gleich.

Wenn wir in Satz 3-1 auch nur gezeigt haben, daß es zu jedem einfachen Pufferautomaten, der sein Verhalten von den beiden ersten Symbolen auf dem Pufferband abhängig macht, einen solchen einfachen Pufferautomaten gibt, der nur das vorderste Symbol berücksichtigt und der mit ihm verwandt ist, so wollen wir im folgenden doch nur noch einfache Pufferautomaten des letzten Typs betrachten.

Es sei nochmals daran erinnert, daß das Eingabeband jeweils um ein Feld bewegt werden muß.

Wir werden zeigen, daß es bzgl. des Annehmens von Mengen keine Einschränkung bedeutet, wenn bei solchen Automaten jeweils nur ein Symbol (evtl. auch die leere Kette Λ) in den Puffer geschrieben werden darf, im Gegensatz zur Möglichkeit des Einschreibens einer endlichen Kette bei den bisher betrachteten einfachen Pufferautomaten.

Satz 3-4: Sei $\mathcal{A} = (X, Z, P, B_2, \delta, z_0)$ ein einfacher Pufferautomat mit den endlichen Mengen X, Z und $P, B_2 := \{b_{20}, b_{21}\}$, $z_0 \in Z$ und der endlichen Korrespondenz δ von $Z \times X \times \bar{P}$ in $Z \times \Omega(P) \times B_2$. Dann gibt es einen einfachen Pufferautomaten $\mathcal{A}' = (X, Z', P', B_2, \delta', z_0)$ mit endlichen nichtleeren Mengen $Z', P', z_0 \in Z'$ und der endlichen Korrespondenz δ' von $Z' \times X \times \bar{P}'$ in $Z' \times \bar{P}' \times B_2$, so daß gilt: $\mathcal{A}$ und $\mathcal{A}'$ sind zueinander äquivalent.

Bew.: Der Beweis wird konstruktiv geführt. Es wird gezeigt, daß durch eine Erweiterung der Zustandsmenge und des Pufferalphabetes das Einschreiben einer Kette in den Puffer durch das Einschreiben eines Symboles aus dem erweiterten Pufferalphabet und das Abarbeiten einer solchen Kette im Puffer durch das Löschen dieses einen Symboles bei gleich-

zeitigem Durchlaufen von gewissen Mengen von Zuständen simuliert werden können.

Der Beweis ist - in dieser Form - nur möglich, weil der Wertebereich endlich sein muß und deshalb die Länge der maximalen Kette, die bei einem Zustandsübergang in den Puffer eingeschrieben wird, bestimmt werden kann.

1) Die Länge dieser größten Kette sei λ . Wir können annehmen, daß $\lambda > 1$ gilt; andernfalls ist nichts zu beweisen. Wir nehmen an, es gebe $v \geq 1$ solche Ketten der Länge λ . Wir werden zeigen, wie sich ein einfacher Pufferautomat $\mathcal{A}'$ konstruieren läßt, der nur noch $v - 1$ Ketten der Länge λ bei einem Zustandsübergang in den Puffer schreibt und der zu dem ursprünglichen Pufferautomaten äquivalent ist. Sei $\pi = p_i \beta p_m$ mit $p_i, p_m \in P$ und $\beta \in \Omega(P)$ eine o.a. Kette der Länge λ . Alle zweiten Komponenten der Werte der Korrespondenz δ , in denen die Kette π auftritt, werden durch ein Symbol q , für das $q \in P$ gilt, ersetzt. Die nach dieser Änderung mit $\bar{\delta}$ bezeichnete Korrespondenz muß noch so erweitert werden, daß die "Verarbeitung" von q möglich wird.

1) Es müssen dazu zunächst alle Relationen betrachtet werden, die das Verhalten des Pufferautomaten in Abhängigkeit von p_i festlegen. Da das Symbol p_i als erstes Symbol der Kette π in den Puffer gelangt sein kann, muß dafür Sorge getragen werden, daß q ebenso behandelt wird, wie dies mit der Kette π in $\mathcal{A}$ geschieht. Gilt

$$(1) \quad \bar{\delta} [z, x, p_i] \supset \{(z_w, \bar{\pi}, b_{21})\} \quad \text{mit } z, z_w \in Z, x \in X, p_i \in P, \\ \bar{\pi} \in \Omega(P_v \{q\}), \text{ so muß } \bar{\delta} \text{ erweitert werden um} \\ \bar{\delta} [z, x, q] \supset \{(\bar{z}'_1, \bar{\pi}, b_{20})\} \quad \text{wobei } \bar{z}'_1 \notin Z.$$

(1) muß in der Korrespondenz enthalten bleiben, da es möglich ist, daß p als Einzelsymbol in den Puffer geschrieben wurde und dann nach (1) verarbeitet werden muß.

Gilt mit $z, z_e \in Z, x \in X, p_i \in P, \bar{\pi} \in \Omega(P \cup \{q\})$ und $b_{20} \in B_2$

(2) $\bar{\delta}[z, x, p_i] \supset \{(z_e, \bar{\pi}, b_{20})\}$,

so muß $\bar{\delta}$ erweitert werden um

$$\bar{\delta}[z, x, q] \supset \{(\bar{z}'_{01}, \bar{\pi}, b_{20})\} \text{ , wobei } \bar{z}'_{01} \notin Z \cup \{\bar{z}'_{02}\} .$$

Es sind alle verschiedenen Folgen zu konstruieren, die das Aussehen haben

$$\begin{aligned} (z^1, \pi^1, b_{20}) &\in \bar{\delta}[z_e, x_e, p_i] \\ (z^2, \pi^2, b_{20}) &\in \bar{\delta}[z^1, x^1, p_i] \\ &\vdots \\ (z^t, \pi^t, b_{20}) &\in \bar{\delta}[z^{t-1}, x^{t-1}, p_i] , \end{aligned}$$

wobei $z^j \in Z, \pi^j \in \Omega(P \cup \{q\})$ für $1 \leq j \leq t$ und $x_e, x^k \in X$ für $1 \leq k \leq t-1$.

Die Konstruktion der verschiedenen Folgen soll abgebrochen werden, wenn gilt:

- (a) $\bar{\delta}[z^t, x, p_i] = \emptyset$ für alle $x \in X$ oder
- (b) $(z^{t+1}, \pi^{t+1}, b_{20}) \in \bar{\delta}[z^t, x^t, p_i]$ oder
- (c) $(z^{t+1}, \pi^{t+1}, b_{20}) \in \bar{\delta}[z^t, x^t, p_i]$ und es gibt ein i mit $1 \leq i \leq t$, so daß gilt: $z^i = z^{t+1}$ und $\pi^i = \pi^{t+1}$.

Diese Konstruktion läßt sich effektiv durchführen: Das Erfülltsein von (a) und (b) läßt sich in trivialer Weise nachprüfen. Sind aber die Bedingungen (a) oder (b) nicht erfüllt, so muß die Folge nach (c) abbrechen, weil $\bar{\delta}$ eine endliche Korrespondenz ist.

Liegen Folgen der Form (3) vor, so muß $\bar{\delta}$ erweitert werden um

$$\begin{aligned} &\bar{\delta}[\bar{z}'_{01}, x_e, q] \supset \{(\bar{z}'_{02}, \pi^1, b_{20})\} \\ (4) \quad &\bar{\delta}[\bar{z}'_{02}, x^1, q] \supset \{(\bar{z}'_{03}, \pi^2, b_{20})\} \\ &\vdots \\ &\bar{\delta}[\bar{z}'_{0,t-1}, x^{t-1}, q] \supset \{(\bar{z}'_{0t}, \pi^t, b_{20})\} \end{aligned}$$

wobei die $\bar{z}_{ij}^1$ ($1 \leq j \leq t$) paarweise verschieden und $\notin Z \cup \{\bar{z}_{ik}^1\}$ sein müssen.

Bricht die Folge (3) nach (a) ab, so ist nichts weiter zu ergänzen, während bei einem Abbrechen nach (b)

$$(4b) \quad \bar{z}_{ot}^1, x^1, q \rightarrow \{(\bar{z}_{ik}^1, \pi^1, b_{20})\}$$

und bei einem Abbrechen nach (c)

$$(4c) \quad \bar{z}_{ot}^1, x^1, q \rightarrow \{(\bar{z}_{o_{i+1}}^1, \pi^1, b_{20})\}$$

hinzuzufügen sind, wobei $\bar{z}_{ik}^1 \notin Z \cup \{\bar{z}_{i1}^1, \bar{z}_{o1}^1, \dots, \bar{z}_{ot}^1\}$.

Für alle nach (3) konstruierbaren Folgen ist die Korrespondenz entsprechend zu erweitern. Es ist dabei darauf zu achten, daß alle neu hinzukommenden Zustände $\bar{z}_{ij}^1$ und $\bar{z}_{ik}^1$ nicht schon in der Zustandsmenge der erweiterten Korrespondenz vorhanden sind, abgesehen von den Fällen der Konstruktion nach (4c).

- 2) Die Symbole $\bar{z}_{ij}^1$ kennzeichnen solche Zustände, in denen gerade das "Abarbeiten" des Symbols p_i im Puffer beim Pufferautomaten $\mathcal{Z}$ erfolgt ist. Da aber in dem neuen zu konstruierenden Pufferautomaten $\mathcal{Z}^1$ die Kette $\pi = p_i \beta p_m$ durch ein Symbol q ersetzt worden war, hat sich das vorderste Symbol im Puffer nicht geändert.

In Abhängigkeit von dem zweiten Symbol der Kette π und mit den $\bar{z}_{ij}^1$ anstelle der z muß der Vorgang nach 1) wiederholt werden. Die dabei erreichten Zustände, die den $\bar{z}_{ij}^1$ des ersten Schrittes entsprechen, werden mit $\bar{z}_{ik}^1$ bezeichnet. Sie dürfen nicht in der Zustandsmenge der erweiterten Korrespondenz enthalten sein.

Analog wird dieser Prozeß der Simulation des Abarbeitens von Symbolen der Kette π weitergeführt, bis Zustände erreicht sind, die denen entsprechen, die der Pufferautomat

$\mathcal{S}$ annimmt, wenn er alle Symbole der Kette π außer dem letzten abgearbeitet hat.

Gibt es in $\mathcal{S}$ eine Eingabefolge $w \in \Omega(X)$, so daß gilt

$$(5) \quad (z_u, w, p; \beta \ p_m \bar{\pi}) \vdash_{\mathcal{S}}^* (z_d, w', p_m \bar{\pi}')$$

mit $z_u, z_d \in Z$, $w, w' \in \Omega(X)$, $p_i, p_m \in P$, $\beta \in \Omega(P)$ und $\bar{\pi}, \bar{\pi}' \in \Omega(P)$, so gilt in $\mathcal{S}'$ nach der bisherigen Konstruktion

$$(z_u, w, q\varphi) \vdash_{\mathcal{S}'}^* (\bar{z}_{u'}^{\lambda-1}, w', q\varphi') \quad \text{mit } \varphi, \varphi' \in \Omega(P \cup \{q\})$$

Gilt dann

$$\bar{\delta} [z_d, x, p_m] \supset \{(z_f, \bar{\pi}, b_{20})\} \quad \text{mit } \bar{\pi} \in \Omega(P \cup \{q\}),$$

so muß $\bar{\delta}$ erweitert werden um

$$\bar{\delta} [\bar{z}_{u'}^{\lambda-1}, x, q] \supset \{(z_f, \bar{\pi}, b_{20})\}.$$

Gilt $\bar{\delta} [z_d, x, p_m] \supset \{(z^{i_1}, \pi^1, b_{20})\}$ mit $\pi^1 \in \Omega(P \cup \{q\})$,

so muß $\bar{\delta}$ erweitert werden um

$$\bar{\delta} [\bar{z}_{u'}^{\lambda-1}, x, q] \supset \{(\bar{z}_{o_1}^{\lambda}, \pi^1, b_{20})\}.$$

Es sind analog zu 1) (3) alle verschiedenen Folgen zu konstruieren, die das Aussehen haben

$$\begin{aligned} (z^{i_2}, \pi^2, b_{20}) &\in \bar{\delta} [z^{i_1}, x^1, p_m] \\ (6) \quad (z^{i_3}, \pi^3, b_{20}) &\in \bar{\delta} [z^{i_2}, x^2, p_m] \\ &\dots \dots \dots \\ (z^{i_e}, \pi^e, b_{20}) &\in \bar{\delta} [z^{i_{e-1}}, x^{e-1}, p_m], \end{aligned}$$

wobei $z^{i_i} \in Z$, $\pi^{i_i} \in \Omega(P \cup \{q\})$ für $1 \leq i \leq e$ und $x^i \in X$ für $1 \leq j \leq e-1$.

Beim Vorliegen solcher Folgen muß $\bar{\delta}$ erweitert werden um

$$\begin{aligned} \bar{\delta} [\bar{z}_{o_1}^{\lambda}, x^1, q] &\supset \{(\bar{z}_{o_2}^{\lambda}, \pi^2, b_{20})\} \\ \bar{\delta} [\bar{z}_{o_2}^{\lambda}, x^2, q] &\supset \{(\bar{z}_{o_3}^{\lambda}, \pi^3, b_{20})\} \\ &\dots \dots \dots \\ \bar{\delta} [\bar{z}_{o_{e-1}}^{\lambda}, x^{e-1}, q] &\supset \{(\bar{z}_{o_e}^{\lambda}, \pi^e, b_{20})\}; \end{aligned}$$

dabei müssen wieder die $\bar{z}_{o_i}^{\lambda}$ ($1 \leq i \leq e$) paarweise verschieden sein und dürfen nicht schon in $\bar{\delta}$ enthalten sein.

Das Abbrechen der Folge nach (3)(a) oder (3)(c) wird analog zu oben behandelt, während bei einem Abbrechen nach (3)(b) in der Form

$$(z_v, \pi^v, b_{\lambda}) \in \bar{\delta}[z', x^e, p_m]$$

die Korrespondenz um

$$\bar{\delta}[\bar{z}_{0,q}^{\lambda}, x^e, q] \supset \{(z_v, \pi^v, b_{\lambda})\}$$

erweitert werden muß.

Diese Erweiterungen müssen für alle z' , für die eine Beziehung in der Art von (5)

$$(z_u, w, p; \beta p_m \bar{\pi}) \vdash_{\mathcal{Z}}^{\lambda} (z', w', p_m \pi')$$

erfüllt ist, vorgenommen werden.

- II) Um zu zeigen, daß $\mathcal{Z}'$ genau die gleiche Menge von Ketten annimmt wie $\mathcal{Z}$, müssen wir das Verhalten von $\mathcal{Z}$ und $\mathcal{Z}'$ unter der Eingabe solcher Folgen aus Eingabesymbolen betrachten, die dazu führen, daß beim Pufferautomaten $\mathcal{Z}$ in den Puffer mindestens einmal die obige Kette π der Länge λ eingeschrieben wird. Geschieht dies nicht, so verhält sich $\mathcal{Z}'$ genau so wie $\mathcal{Z}$, da $\bar{\delta}$ eine Erweiterung von δ ist.

Wir brauchen nur solche Konfigurationen zu berücksichtigen, in denen das vorderste Symbol im Puffer gleich dem ersten Symbol der Kette π ist. Die verschiedenen Möglichkeiten, nach denen π in $\mathcal{Z}$ verarbeitet wird und die wegen der Nichtdeterminiertheit auch nebeneinander auftreten können, müssen diskutiert werden:

- 1) Mit $x^i \in X$ ($1 \leq i \leq n$), $p_i, p_m \in P$, $\beta, \pi^k \in Q(P)$ ($0 \leq k \leq n-f$) gelte:

$$\begin{aligned} (z_k, x^f \dots x^n, p_i; \beta p_m \pi^0) \vdash_{\mathcal{Z}} (z_l, x^{f+1} \dots x^n, p_i; \beta p_m \pi^0 \pi^1) \vdash_{\mathcal{Z}}^{\lambda} \\ \vdash_{\mathcal{Z}}^{\lambda} (z_m, x^n, p_i; \beta p_m \pi^0 \pi^1 \dots \pi^{n-f}) \vdash_{\mathcal{Z}} (z_r, \wedge, p_i; \beta p_m \pi^0 \pi^1 \dots \pi^{n-f+1}) \end{aligned}$$

Nach der Konstruktion von $\mathcal{L}'$ gilt dann:

$$\begin{aligned} (z_k, x^f \dots x^n, q\alpha'(\pi^0)) \vdash_{\mathcal{L}'}^* (\bar{z}_{oi}^f, x^{f+i} \dots x^n, q\alpha'(\pi^0)\alpha(\pi^i)) \vdash_{\mathcal{L}'}^* \\ \vdash_{\mathcal{L}'}^* (\bar{z}_{oj}^f, x^n, q\alpha'(\pi^0) \dots \alpha(\pi^{n-f})) \vdash_{\mathcal{L}'}^* \\ \vdash_{\mathcal{L}'}^* (\bar{z}_{ou}^f, \wedge, q\alpha'(\pi^0) \dots \alpha(\pi^{n-f+i})) \end{aligned}$$

Dabei bedeutet $\alpha(\varphi)$ die Pufferinschrift in $\mathcal{L}'$, falls in $\mathcal{L}$ φ in den Puffer geschrieben wurde.

Es gilt:

$$\alpha(\varphi) = \begin{cases} q, \text{ falls } \varphi = \pi \\ \varphi \text{ sonst} \end{cases}$$

$\alpha'(\pi^0)$ sei die entsprechend der Reihenfolge und der Form des Einschreibens von π^0 gebildete Kette.

2) Sei p_v ein Symbol der Teilkette βp_m von π :

$$\beta p_m = \beta' p_v \beta'' \quad \text{mit } p_m, p_v \in P \text{ und } \beta, \beta', \beta'' \in \Omega(P).$$

Es gelte:

$$\begin{aligned} (z_k, x^f \dots x^n, p_i \beta' p_v \beta'' \pi^0) \vdash_{\mathcal{L}'}^* (z_s, x^s \dots x^n, p_v \beta'' \pi^0 \pi^i \dots \pi^{s-f}) \\ \text{und } \delta[z_s, x^s, p_v] = \emptyset, \text{ d.h. die betreffende Kette, von} \\ \text{der } x^f \dots x^n \text{ eine Teilkette ist, wird in } \mathcal{L} \text{ nicht an-} \\ \text{genommen.} \end{aligned}$$

Nach der Konstruktion von $\mathcal{L}'$ gilt dann:

$$\begin{aligned} (z_k, x^f \dots x^n, q\alpha'(\pi^0)) \vdash_{\mathcal{L}'}^* (\bar{z}_{im}^f, x^s \dots x^n, q\alpha'(\pi^0) \dots \alpha(\pi^{s-f})) \\ \text{und } \bar{\delta}[\bar{z}_{im}^f, x^s, q] = \emptyset. \end{aligned}$$

3) Mit den Bedingungen von 2) gelte

$$\begin{aligned} (z_k, x^f \dots x^n, p_i \beta' p_v \beta'' \pi^0) \vdash_{\mathcal{L}'}^* (z_l, x^{f+i} \dots x^n, p_i \beta' p_v \beta'' \pi^0 \pi^i) \vdash_{\mathcal{L}'}^* \\ \vdash_{\mathcal{L}'}^* (z_r, x^r \dots x^n, \beta' p_v \beta'' \pi^0 \pi^i \dots \pi^{r-f}) \vdash_{\mathcal{L}'}^* \\ \vdash_{\mathcal{L}'}^* (z_s, x^s \dots x^n, p_v \beta'' \pi^0 \pi^i \dots \pi^{s-f}) \vdash_{\mathcal{L}'}^* \\ \vdash_{\mathcal{L}'}^* (z_t, x^n, p_v \beta'' \pi^0 \pi^i \dots \pi^{n-f}) \vdash_{\mathcal{L}'}^* (z_u, \wedge, p_v \beta'' \pi^0 \pi^i \dots \pi^{n-f+i}) \end{aligned}$$

Nach der Konstruktion von $\mathcal{L}'$ gilt dann:

$$\begin{aligned} (z_k, x^f \dots x^n, q\alpha'(\pi^0)) \vdash_{\mathcal{L}'}^* (\bar{z}_{oi}^f, x^{f+i} \dots x^n, q\alpha'(\pi^0)\alpha(\pi^i)) \vdash_{\mathcal{L}'}^* \\ \vdash_{\mathcal{L}'}^* (\bar{z}_{oj}^f, x^r \dots x^n, q\alpha'(\pi^0) \dots \alpha(\pi^{r-f})) \vdash_{\mathcal{L}'}^* \\ \vdash_{\mathcal{L}'}^* (\bar{z}_{om}^f, x^s \dots x^n, q\alpha'(\pi^0) \dots \alpha(\pi^{s-f})) \vdash_{\mathcal{L}'}^* \\ \vdash_{\mathcal{L}'}^* (\bar{z}_{ou}^f, x^n, q\alpha'(\pi^0) \dots \alpha(\pi^{n-f})) \vdash_{\mathcal{L}'}^* (\bar{z}_{ov}^f, \wedge, q\alpha'(\pi^0) \dots \alpha(\pi^{n-f+i})) \end{aligned}$$

$$\begin{aligned}
 4) \quad & (z_k, x^f \dots x^n, p_i \beta p_m \pi^o) \vdash_{\mathcal{Z}}^* (z_r, x^r \dots x^n, \beta p_m \pi^o \pi^1 \dots \pi^{r-f}) \\
 & \vdash_{\mathcal{Z}}^* (z_s, x^s \dots x^n, p_m \pi^o \pi^1 \dots \pi^{s-f}) \\
 & \vdash_{\mathcal{Z}}^* (z_t, x^t \dots x^n, p_m \pi^o \pi^1 \dots \pi^{t-f}) \\
 & \vdash_{\mathcal{Z}} (z_t, x^t \dots x^n, \pi^o \pi^1 \dots \pi^{t-f})
 \end{aligned}$$

Nach der Konstruktion von $\mathcal{Z}'$ gilt dann :

$$\begin{aligned}
 & (z_k, x^f \dots x^n, q\alpha'(\pi^o)) \vdash_{\mathcal{Z}'}^* (\bar{z}_{4i}^1, x^r \dots x^n, q\alpha'(\pi^o) \dots \alpha(\pi^{r-f})) \\
 & \vdash_{\mathcal{Z}'}^* (\bar{z}_{4m}^1, x^s \dots x^n, q\alpha'(\pi^o) \dots \alpha(\pi^{s-f})) \\
 & \vdash_{\mathcal{Z}'}^* (\bar{z}_{0u}^1, x^t \dots x^n, q\alpha'(\pi^o) \dots \alpha(\pi^{t-f})) \\
 & \vdash_{\mathcal{Z}'} (z_t, x^t \dots x^n, \alpha'(\pi^o) \dots \alpha(\pi^{t-f}))
 \end{aligned}$$

Werden Eingabeketten in $\mathcal{Z}$ nach 1), 2) oder 3) bearbeitet, so bedeutet dies, daß sie nicht angenommen werden, weil keine Konfiguration ¹⁾ $(z, \wedge, \wedge)$ mit $z \in Z$ erreicht worden ist.

Dasselbe gilt für diese Ketten im Pufferautomaten $\mathcal{Z}'$. Auch dort ist keine Konfiguration $(z', \wedge, \wedge)$ mit z' aus der erweiterten Zustandsmenge erreichbar, und damit sind die Ketten auch von $\mathcal{Z}'$ nicht angenommen worden und umgekehrt.

Durch vollständige Induktion über die Länge der Übergangsfolgen ergibt sich, daß die Mengen der angenommenen Eingabeketten, deren Verarbeitung also nach 4) verlaufen muß, für $\mathcal{Z}$ und $\mathcal{Z}'$ übereinstimmen.

Durch Fortsetzen dieses Verfahrens für alle übrigen Ketten der Länge λ und für die Ketten, die eine größere Länge als 1 haben und bei einem Zustandsübergang in den Puffer eingeschrieben werden, ergibt sich die Behauptung.

Korollar: Die Klasse der von einfachen Pufferautomaten mit der endlichen Korrespondenz $\delta: Z \times X \times \bar{P} \rightarrow Z \times \Omega(P) \times B_2$ angenommenen Mengen und die Klasse, der von einfachen Pufferautomaten mit der endlichen Korrespondenz $\delta': Z' \times X' \times \bar{P}' \rightarrow Z' \times \bar{P}' \times B_2$ angenommenen Mengen stimmen überein.

¹⁾ Bei Pufferautomaten werden wir Elemente von $Z \times \Omega(X) \times \Omega(P)$ auch Konfigurationen nennen.

Dies folgt aus dem obigen Satz und der Tatsache, daß einfache Pufferautomaten mit der endlichen Korrespondenz δ' in der Klasse der einfachen Pufferautomaten mit der endlichen Korrespondenz δ enthalten sind.

Wir wollen ein Beispiel für die Überführung eines Pufferautomaten des Typs $\mathcal{Z}$ von oben in einen solchen des Typs $\mathcal{Z}'$ angeben:

Sei h ein Homomorphismus von $\Omega(\{a,b,c\})$ in $\Omega(\{e,f,g\})$ mit $h(\Lambda) = \Lambda, h(a) = ef, h(b) = eg, h(c) = f$ und $h(x' \dots x^k) = h(x') \dots, \dots, h(x^k)$ mit $x^i \in \{a,b,c\}$ und $k \in \mathbb{N}$. Sei $L := \{w \mid w \in \Omega(\{a,b,c\}) \setminus \{\Lambda\}\}$

Ein Beispiel für einen einfachen Pufferautomaten, der diese Sprache L annimmt, ist das folgende:

$\mathcal{Z} = (X, Z, P, B_2, \delta, z_0)$, wobei $X := \{a,b,c,e,f,g,t\}$, $Z := \{z_0, z_1, z_2\}$, $P := \{e,f,g\}$, $B_2 := \{b_{20}, b_{21}\}$ und δ sei eine endliche Korrespondenz $\delta : Z \times X \times \bar{P} \rightarrow Z \times \Omega(P) \times B_2$, die folgendermaßen festgelegt ist:

$$\begin{aligned} \delta[z_0, a, \Lambda] &= \{(z_1, ef, b_{20})\} \\ \delta[z_0, b, \Lambda] &= \{(z_1, eg, b_{20})\} \\ \delta[z_0, c, \Lambda] &= \{(z_1, f, b_{20})\} \\ (\forall p/p \in P) (\delta[z_1, a, p] &= \{(z_1, ef, b_{20})\}) \\ (\forall p/p \in P) (\delta[z_1, b, p] &= \{(z_1, eg, b_{20})\}) \\ (\forall p/p \in P) (\delta[z_1, c, p] &= \{(z_1, f, b_{20})\}) \\ (\forall p/p \in P) (\delta[z_1, t, p] &= \{(z_2, \Lambda, b_{20})\}) \\ \delta[z_2, e, e] &= \{(z_2, \Lambda, b_{21})\} \\ \delta[z_2, f, f] &= \{(z_2, \Lambda, b_{21})\} \\ \delta[z_2, g, g] &= \{(z_2, \Lambda, b_{21})\} \end{aligned}$$

Man macht sich leicht klar, daß dieser Pufferautomat gerade die Sätze der oben angegebenen Sprache annimmt.

Nach dem oben Gesagten konstruieren wir den zu $\mathcal{Z}$ äquivalenten

Pufferautomaten $\mathcal{Z}' = (X, Z', P', B_2, \delta', z_0)$, wobei $Z' := Z \cup \{\bar{z}_1', \bar{z}_2'\}$,
 $P' := P \cup \{q_1, q_2\}$ und die endliche Korrespondenz $\delta' : Z' \times X \times \bar{P}'$
 $\rightarrow Z' \times \bar{P}' \times B_2$ wie unten beschrieben aussieht.

Es sei $\alpha(ef) = q_1$, $\alpha(eg) = q_2$ und $\alpha(f) = f$.

$$(1) \delta'[z_0, a, \wedge] = \{(z_1, q_1, b_{20})\}$$

$$(2) \delta'[z_0, b, \wedge] = \{(z_1, q_2, b_{20})\}$$

$$(3) \delta'[z_0, c, \wedge] = \{(z_1, f, b_{20})\}$$

$$(4) (\forall p'/p' \in P') (\delta'[z_1, a, p'] = \{(z_1, q_1, b_{20})\})$$

$$(5) (\forall p'/p' \in P') (\delta'[z_1, b, p'] = \{(z_1, q_2, b_{20})\})$$

$$(6) (\forall p'/p' \in P') (\delta'[z_1, c, p'] = \{(z_1, f, b_{20})\})$$

$$(7) (\forall p'/p' \in P') (\delta'[z_1, t, p'] = \{(z_2, \wedge, b_{20})\})$$

$$(8) \delta'[z_2, e, e] = \{(z_2, \wedge, b_{21})\}$$

$$(9) \delta'[z_2, e, q_1] = \{(\bar{z}_1', \wedge, b_{20})\}$$

$$(10) \delta'[z_2, e, q_2] = \{(\bar{z}_2', \wedge, b_{20})\}$$

$$(11) \delta'[z_2, f, f] = \{(z_2, \wedge, b_{21})\}$$

$$(12) \delta'[z_2, g, g] = \{(z_2, \wedge, b_{21})\}$$

$$(13) \delta'[\bar{z}_1', f, q_1] = \{(z_2, \wedge, b_{21})\}$$

$$(14) \delta'[\bar{z}_2', g, q_2] = \{(z_2, \wedge, b_{21})\}$$

Man sieht, daß man P' auf die Menge $\{f, q_1, q_2\}$ beschränken und auf die Zeilen (8) und (12) verzichten könnte, daß dadurch aber auch keine Änderungen an der angenommenen Menge $T(\mathcal{Z}')$ bewirkt werden.

Nach Satz 3-4 können wir uns also künftig auf die Betrachtung solcher einfacher Pufferautomaten beschränken, bei denen jeweils nur ein Symbol in den Puffer eingeschrieben werden darf. Falls nichts anderes gesagt, wollen wir dies auch tun, jedoch wird es für Beispiele manchmal bequemer sein, auf diese Einschränkung zu verzichten.

4. Einordnung der deterministischen Puffersprachen in die Typenhierarchie für formale Sprachen

In diesem Abschnitt soll gezeigt werden, wie sich die noch zu definierenden deterministischen Puffersprachen in die Hierarchie der formalen Sprachen nach CHOMSKY (s.z.B. CHOMSKY(1959)) einfügen. Als wichtigstes Ergebnis werden wir dabei erhalten, daß sie nicht mit einem bestimmten Typ von Sprachen identifizierbar sind.

Wie es auch bei den anderen Automatentypen üblich ist, wollen wir kurz auf den Unterschied zwischen deterministischen und nichtdeterministischen einfachen Pufferautomaten eingehen. Die einfachen Pufferautomaten aus Abschnitt 3. waren nicht-deterministisch und unvollständig, d.h. bei einer vorgegebenen Konfiguration konnte es für den Automaten keine, eine oder mehrere Möglichkeiten geben, seine Arbeit fortzusetzen. Der einfache deterministische Pufferautomat ist dadurch gekennzeichnet, daß es für ihn jeweils genau eine Möglichkeit gibt weiterzuarbeiten.

Def.: Ein einfacher deterministischer Pufferautomat ist ein Sextupel $\mathcal{S}_D = (X, Z, P, B_2, d, z_0)$. Dabei sind X, Z und P endliche, nichtleere Mengen, $B_2 := \{b_{20}, b_{21}\}$, $z_0 \in Z$, und d ist eine Abbildung von $Z \times X \times \bar{P}$ in $Z \times \bar{P} \times B_2$.

Die Definitionen der Relation " $\vdash$ " und des Annehmens von Wörtern durch den einfachen deterministischen Pufferautomaten wollen wir nicht explizit angeben, da sie analog zu den entsprechenden Definitionen für den einfachen Pufferautomaten in Abschnitt 3. formuliert werden. Wir können nun sagen, was wir unter Puffersprachen verstehen wollen.

Def.: Die Menge der von einem einfachen deterministischen Pufferautomaten angenommenen Wörter wird als deterministische Puffersprache bezeichnet.

Die Menge der von einem einfachen Pufferautomaten angenommenen Wörter wird als nichtdeterministische Puffersprache bezeichnet. Falls wir nur von Puffersprachen reden, seien darunter nichtdeterministische Puffersprachen verstanden.

Satz 4-1: (a) Jede deterministische Puffersprache ist zugleich eine nichtdeterministische Puffersprache.

(b) Das Umgekehrte gilt nicht.

Bew.: Nach obigen Definitionen ist der Teil (a) des Satzes klar. Jeder einfache deterministische Pufferautomat ist ein Spezialfall eines einfachen Pufferautomaten.

(b) Am Beispiel der Sprache $L := \{ww \mid w \in \Omega(\{a,b\})^*\}$ soll (b) bewiesen werden.

(1) Wir machen die Annahme, es gebe einen einfachen deterministischen Pufferautomaten $\mathcal{Z}_D$, so daß gilt:

$$T(\mathcal{Z}_D) = L.$$

Da die Zustandsmenge Z endlich ist, gibt es zwei Ketten a^{2i} und a^{2j} mit $i, j \in \mathbb{N}, i \neq j$ und einen Zustand z , so daß gilt:

$$(z_0, a^{2i}, \wedge) \vdash_{\mathcal{Z}_D}^* (z, \wedge, \wedge)$$

$$\text{und } (*)(z_0, a^{2j}, \wedge) \vdash_{\mathcal{Z}_D}^* (z, \wedge, \wedge).$$

- a^{2i} und a^{2j} sind ja Elemente von L .

$a^{2i}b^1a^{2i}b^1$ ist ebenfalls ein Element von L , d.h.

es gilt

$$(z_0, a^{2i}b^1a^{2i}b^1, \wedge) \vdash_{\mathcal{Z}_D}^* (z, b^1a^{2i}b^1, \wedge) \vdash_{\mathcal{Z}_D}^* (z', \wedge, \wedge).$$

Wegen der Determiniertheit des Automaten $\mathcal{A}_D$ gilt aber auch nach (*)

$$(z_0, a^{2j} b^1 a^{2i} b^1, \Lambda) \vdash_{\mathcal{A}_D}^* (z, b^1 a^{2i} b^1, \Lambda) \vdash_{\mathcal{A}_D}^* (z', \Lambda, \Lambda),$$

d.h. $a^{2j} b^1 a^{2i} b^1$ ist ein Element aus $T(\mathcal{A}_D)$, aber wegen $i \neq j$ kein Element aus L . Damit ist obige Annahme zum Widerspruch geführt.

- (2) Sei $\mathcal{A}$ der folgendermaßen gegebene einfache Pufferautomat: $\mathcal{A} = (\{a, b\}, \{z_0, z_1, z_2\}, \{a, b\}, \{b_{20}, b_{21}\}, \delta, z_0)$ und δ ist bestimmt durch

$$\begin{aligned} \delta[z_0, a, \Lambda] &= \{(z_1, a, b_{20})\} \\ \delta[z_0, b, \Lambda] &= \{(z_1, b, b_{20})\} \\ \delta[z_1, a, a] &= \{(z_1, a, b_{20}), (z_2, \Lambda, b_{21})\} \\ \delta[z_1, a, b] &= \{(z_1, a, b_{20})\} \\ \delta[z_1, b, a] &= \{(z_1, b, b_{20})\} \\ \delta[z_1, b, b] &= \{(z_1, b, b_{20}), (z_2, \Lambda, b_{21})\} \\ \delta[z_2, a, a] &= \{(z_2, \Lambda, b_{21})\} \\ \delta[z_2, b, b] &= \{(z_2, \Lambda, b_{21})\} \end{aligned}$$

Man sieht unmittelbar, daß gilt: $T(\mathcal{A}) = L$.

Damit ist gezeigt, daß es eine Sprache gibt, die von einem einfachen Pufferautomaten, nicht aber von einem einfachen deterministischen Pufferautomaten angenommen wird.

Um die Puffersprachen gegenüber den anderen formalen Sprachen abzugrenzen, ist es bequem, die "entsprechenden" Automatentypen zu betrachten. Wir benötigen zunächst den Begriff des endlichen, erkennenden Automaten.

Def.: Ein endlicher erkennender Automat ist ein Quintupel

$$\mathcal{A} = (X, Z, \alpha, z_0, F), \text{ wobei } X \text{ und } Z \text{ endliche, nichtleere}$$

Mengen sind, $F \subset Z$, $z_0 \in Z$ und α eine Abbildung von $Z \times X$ in Z ist.

Bekanntlich gibt es zu jeder regulären Sprache einen endlichen erkennenden Automaten, der sie annimmt. Eine Kette $w \in Q(X)$ heißt dabei vom Automaten $\mathcal{A}$ angenommen, wenn sich der Automat nach dem Einlesen von w in einem markierten Zustand, d.h. in einem Zustand aus F befindet. Die Menge der so angenommenen Ketten wird als die vom Automaten angenommene Sprache bezeichnet. (genaue Definitionen s. HÄNDLER (1966))

Über das Verhältnis der regulären Sprachen zu den Puffersprachen gibt der folgende Satz Auskunft:

Satz 4-2: (a) Jede reguläre Sprache ist eine deterministische Puffersprache.

(b) Das Umgekehrte gilt nicht.

Bew.: (a) Der Beweis wird über die zugehörigen Automaten geführt.

Sei L eine reguläre Sprache und $\mathcal{A} = (X, Z, d, z_0, F)$ sei ein endlicher erkennender Automat, der sie annimmt.

Dann sei $\mathcal{L}_D = (X, Z, \{p\}, \{b_{20}, b_{21}\}, d', z_0)$ ein deterministischer Pufferautomat mit der folgendermaßen definierten Abbildung:

Für alle $(z, x) \in Z \times X$ werde gesetzt:

$$\left. \begin{aligned} d'(z, x, \Lambda) &= (z', p, b_{20}) \\ d'(z, x, p) &= (z', \Lambda, b_{20}) \end{aligned} \right\} \text{ falls } d(z, x) = z' \wedge z' \in Z \setminus F$$

$$\left. \begin{aligned} d'(z, x, \Lambda) &= (z'', \Lambda, b_{20}) \\ d'(z, x, p) &= (z'', \Lambda, b_{21}) \end{aligned} \right\} \text{ falls } d(z, x) = z'' \wedge z'' \in F$$

$\mathcal{L}_D$ weist dieselben Zustandsübergänge auf wie $\mathcal{A}$, schreibt das Symbol p in den Puffer, bzw. läßt es ungeändert stehen, falls $\mathcal{A}$ in einen nichtmarkierten Zustand übergeht und löscht den Puffer, bzw. läßt

ihn leer, falls $\mathcal{L}$ in einen markierten Zustand übergeht. Es ist unmittelbar ersichtlich, daß gilt:

$$T(\mathcal{S}_D) = L.$$

- (b) Die Menge $L := \{a^n b a^n \mid n \in \mathbb{N}\}$ ist bekanntlich keine reguläre Sprache (s.z.B. HÄNDLER (1966)).

Wir wollen nun einen einfachen deterministischen Pufferautomaten $\mathcal{S}_D$ angeben, der gerade diese Sprache annimmt:

$$\mathcal{S}_D = (\{a, b\}, \{z_0, z_1, z_2, z_3\}, \{p\}, \{b_{20}, b_{21}\}, d, z_0)$$

$$\begin{aligned} \text{mit } d(z_0, a, \wedge) &= (z_1, p, b_{20}) \\ d(z_0, b, \wedge) &= (z_3, p, b_{20}) \\ d(z_0, a, p) &= (z_3, p, b_{20}) \\ d(z_0, b, p) &= (z_3, p, b_{20}) \\ d(z_1, a, \wedge) &= (z_3, p, b_{20}) \\ d(z_1, b, \wedge) &= (z_3, p, b_{20}) \\ d(z_1, a, p) &= (z_1, p, b_{20}) \\ d(z_1, b, p) &= (z_2, \wedge, b_{20}) \\ d(z_2, a, \wedge) &= (z_3, p, b_{20}) \\ d(z_2, b, \wedge) &= (z_3, p, b_{20}) \\ d(z_2, a, p) &= (z_2, \wedge, b_{21}) \\ d(z_2, b, p) &= (z_3, p, b_{20}) \\ d(z_3, a, \wedge) &= (z_3, p, b_{20}) \\ d(z_3, b, \wedge) &= (z_3, p, b_{20}) \\ d(z_3, a, p) &= (z_3, p, b_{20}) \\ d(z_3, b, p) &= (z_3, p, b_{20}) \end{aligned}$$

Man macht sich leicht klar, daß $T(\mathcal{S}_D) = L$.

Im Teil (a) des Beweises mußte der Puffer mitbenutzt werden, um $\mathcal{L}$ zu simulieren, da es reguläre Sprachen gibt - z.B.

$\{(ab)^n \mid n \in \mathbb{N}\}$ -, die nicht von einem endlichen erkennenden Automaten angenommen werden, für den $F = Z$ gilt.

Nachdem wir mit den regulären Sprachen gleichsam eine "untere Grenze" für die Puffersprachen in der Typenhierarchie von CHOMSKY gefunden haben, versuchen wir nun, sie "nach oben" abzugrenzen. Dazu benötigen wir noch den Begriff des linear-beschränkten Automaten (LANDWEBER (1963), KURODA (1964), GINSBURG/ROSE (1966)).

Def.: Ein linear-beschränkter Automat ist ein Quintupel

$\mathcal{A} = (X, Z, \delta, z_0, F)$, wobei X und Z endliche, nichtleere, disjunkte Mengen sind, $z_0 \in Z$, $F \subseteq Z$ gilt und δ eine Korrespondenz von $Z \times X$ in $Z \times X \times \{-1, 0, 1\}$ ist.

Die Wirkungsweise des linear-beschränkten Automaten wollen wir in folgender Notation beschreiben:

Gegeben sei der obige Automat $\mathcal{A}$. Über $\Omega(X)Z\Omega(X)$ sei folgendermaßen eine Relation $\vdash^*$ definiert:

Seien $z, z' \in Z, r, x, y \in X, u, v \in \Omega(X)$. Dann gelte

$(uxzyv) \vdash_{\mathcal{A}} (uz'xrv)$ falls $(z', r, -1) \in \delta[z, y]$
 $(uzyv) \vdash_{\mathcal{A}} (uz'rv)$ falls $(z', r, 0) \in \delta[z, y]$
 $(uxzyv) \vdash_{\mathcal{A}} (urz'yv)$ falls $(z', r, 1) \in \delta[z, x]$.

Für alle $(uzv) \in \Omega(X)Z\Omega(X)$ gelte $(uzv) \vdash_{\mathcal{A}}^* (uzv)$.

Für $(uzv), (u'z'v') \in \Omega(X)Z\Omega(X)$ gelte $(uzv) \vdash_{\mathcal{A}}^* (u'z'v')$, falls $(uzv) = (u^1z^1v^1), (u^2z^2v^2), \dots, (u^kz^kv^k) = (u'z'v')$ existieren, so daß gilt $(u^i z^i v^i) \vdash_{\mathcal{A}} (u^{i+1} z^{i+1} v^{i+1})$ für $1 \leq i \leq k-1$.

Es läßt sich mit z.B. $(uxzyv) \vdash_{\mathcal{A}} (uz'xrv)$ folgende anschauliche Vorstellung verbinden: Der linear-beschränkte Automat befindet sich im Zustand z und geht beim Lesen des Symbol y in den Zustand z' über, ersetzt das Symbol y durch r und bewegt den Leseschreibkopf um ein Feld nach links.

Der linear-beschränkte Automat soll ebenso wie der Pufferautomat in seiner Eigenschaft als erkennender Automat benutzt werden. Wir müssen deshalb definieren, unter welchen Umständen

Untermengen einer freien Halbgruppe über einem Alphabet angenommen werden.

Def.: Sei $\mathcal{A} = (X, Z, \delta, z_0, F)$ ein linear-beschränkter Automat.

Eine Kette $w \in \Omega(X)$ heißt vom linear-beschränkten Automaten $\mathcal{A}$ angenommen, wenn gilt

$$(z_0, w) \vdash_{\mathcal{A}}^* (uz) \quad \text{für irgendein } (uz) \in \Omega(X)F.$$

Die Menge der vom linear-beschränkten Automaten $\mathcal{A}$ angenommenen Ketten $w \in \Omega(X)$ wird mit $T(\mathcal{A})$ bezeichnet.

Ein Wort wird also angenommen, wenn der Automat in einen markierten Zustand gelangt und dabei eine Konfiguration vorliegt, in der rechts vom Leseschreibkopf keine Symbole aus X auftreten.

Lemma 4-1: Zu jedem einfachen Pufferautomaten

$$\mathcal{B} = (X^*, Z^*, P^*, B_2, \delta^*, z_0^*), \text{ wobei } X^*, Z^*, P^* \text{ endliche, nichtleere Mengen sind, } z_0^* \in Z^*, B_2 := \{b_{20}, b_{21}\} \text{ und } \delta^* \text{ eine endliche Korrespondenz von } Z^* \times X^* \times \bar{P}^* \text{ in } Z^* \times \Omega(P^*) \times B_2 \text{ ist, gibt es einen linear-beschränkten Automaten } \mathcal{A} = (X, Z, \delta, z_0, F), \text{ so daß gilt:}$$

$$T(\mathcal{A}) \setminus \{\Lambda\} = T(\mathcal{B}).$$

Bew.: Nach Satz 3-4 gibt es zu $\mathcal{B}$ einen Pufferautomaten

$$\mathcal{B}' = (X', Z', P', B_2, \delta', z_0') \text{ mit } \delta': Z' \times X' \times \bar{P}' \rightarrow Z' \times \bar{P}' \times B_2, \text{ so daß } \mathcal{A} \text{ und } \mathcal{B}' \text{ zueinander äquivalent sind.}$$

Es genügt also, zu zeigen, daß es zu jedem solchen Pufferautomaten $\mathcal{B}'$ einen linear-beschränkten Automaten $\mathcal{A}$ gibt, so daß gilt: $T(\mathcal{A}) \setminus \{\Lambda\} = T(\mathcal{B}')$. Der Beweis wird durch die Konstruktion des entsprechenden linear-beschränkten Automaten $\mathcal{A}$ erbracht.

Sei $\mathcal{B}'$ obiger Pufferautomat. Folgende Mengen werden definiert:

$$P_A := P' \times \{A\} ; \quad P_E := P' \times \{E\} ;$$

$P_{AE} := P' \times \{AE\} \cup \{\lambda\}$, wobei P' , $\{A\}$, $\{E\}$, $\{AE\}$ und $\{\lambda\}$ paarweise disjunkt sind. Die Elemente aus den definierten Mengen sollen mit resp. p^A, p^E, p^{AE} bzw. λ abgekürzt bezeichnet werden.

Sei $\mathcal{L} = (X, Z, \delta, z_0, P')$ ein linear-beschränkter Automat

mit $X := X' \cup P' \cup P_A \cup P_E \cup P_{AE} \cup \{1\}$,

wobei $1 \notin X' \cup P' \cup P_A \cup P_E \cup P_{AE}$,

$Z := Z' \cup \{Z' \times X' \times \bar{P}' \times Z' \times \bar{P}' \times B_2 \times \{1, 2, 3\}\} \cup \{Z' \times Z'\}$,
 $z_0 \in Z$ und $P := \{Z' \times Z'\}$.

Die Konstruktion von δ soll weiter unten angegeben werden, nachdem die Vorgehensweise skizziert wurde:

Die Kette $w \in \Omega(X')$, die in $\mathcal{L}'$ auf dem Eingabeband vorliegt, ist in $\mathcal{L}$ auf dem (einzigen) Arbeitsband enthalten. In $\mathcal{L}'$ wird bei jedem Zustandsübergang ein Symbol der Eingabekette gelesen und danach nicht mehr betrachtet. Bei jedem Zustandsübergang kann aber auch höchstens ein Symbol in den Puffer geschrieben werden. Die Länge der gesamten Eingabekette in $\mathcal{L}'$ und in $\mathcal{L}$ ist immer größer als die oder gleich der Summe der Pufferlänge und der Länge der relevanten Eingabekette. Damit ist eine wesentliche Voraussetzung dafür erfüllt, daß $\mathcal{L}'$ mit dem linear-beschränkten Automaten $\mathcal{L}$ simuliert werden kann. Der Pufferinhalt läßt sich auf den nicht mehr (als Eingabeinformation) relevanten Teil des Arbeitsbandes von $\mathcal{L}$ notieren. Ist der Pufferinhalt in $\mathcal{L}'$ leer, so wird dies in $\mathcal{L}$ durch ein spezielles Symbol λ festgelegt, enthält er in $\mathcal{L}'$ genau ein Symbol p , so steht auf dem erwähnten Teil des Arbeitsbandes in $\mathcal{L}$ das Symbol p^{AE} , enthält dagegen der Puffer in $\mathcal{L}'$ eine Kette der Länge > 1 , so ist diese Kette auch auf dem Arbeitsband

notiert, mit der Änderung, daß das Symbol p_k des Puffers, zu dem $\mathcal{Z}'$ Zugriff hat und das in $\mathcal{Z}$ am weitesten links steht durch das entsprechende Symbol p_k^A ersetzt wird, während das zuletzt in den Puffer eingeschriebene Symbol p_j in $\mathcal{Z}$ durch p_j^E ersetzt wird. p_j^E steht auf dem Band rechts von p_k^A .

Dem Einlesen eines Symboles vom Eingabeband beim Pufferautomaten $\mathcal{Z}'$ und dem Übergang in einen bestimmten Zustand in Abhängigkeit von dem bisherigen Zustand, dem gelesenen Eingabesymbol und dem vordersten Puffersymbol entspricht beim linear-beschränkten Automaten das Lesen des entsprechenden Symbols auf dem Arbeitsband, das Ersetzen dieses Symbols durch ein spezielles Symbol l , ein Linkslaufen des Leseschreibkopfes über eine für den bisherigen Zustand charakteristische Folge von Zuständen, die $\notin Z'$, bis zu dem (einzigen) Element $p_1^A \in P_A$ bzw. $p_1^{AE} \in P_{AE}$, das auf dem Arbeitsband enthalten ist. Dies ist das Analogon zu dem vordersten Puffersymbol p_1 . Dieser Teil des Arbeitsbandes wird entsprechend der Vorgehensweise von $\mathcal{Z}'$ behandelt: Wird ein Symbol im Puffer gelöscht (d.h. in der Konfiguration steht b_{21}), so bedeutet dies, daß p_k^A gelöscht wird und das rechts davon notierte Symbol p_l in p_l^A geändert wird. Falls jedoch das zu ändernde Symbol aus P_E ist und bei dem betreffenden Zustandsübergang in $\mathcal{Z}'$ kein weiteres Symbol in den Puffer eingeschrieben wird, so ist es durch das entsprechende Symbol aus P_{AE} zu ersetzen. Ist im Puffer von $\mathcal{Z}'$ nur noch ein Symbol vorhanden, so bedeutet dies, daß es in $\mathcal{Z}$ durch ein Symbol aus $P_{AE} \setminus \{\lambda\}$ notiert ist. Wird das Symbol im Puffer gelöscht, ohne daß bei diesem Zustandsübergang

gang ein Symbol aus P' eingeschrieben wird, so wird in $\mathcal{L}$ das entsprechende Symbol aus $P_{AE} \setminus \{\lambda\}$ durch das Symbol λ ersetzt, und es wird in einen Zustand (z_k, z_k) aus F übergegangen, wenn der entsprechende Übergang in $\mathcal{L}'$ nach z_k erfolgt wäre. Unter Beibehaltung des Zustandes (z_k, z_k) werden beim Gang nach rechts die Symbole, die nicht aus X' sind, überlesen. Sind keine Symbole aus X' mehr auf dem Arbeitsband enthalten, ist also die Eingabekette schon vollständig abgearbeitet, so bleibt der Leseschreibkopf hinter dem letzten Feld der Eingabekette stehen. Da der linear-beschränkte Automat in einem Zustand aus F ist, bedeutet dies, daß die Kette als angenommen gilt, was ja auch verlangt werden muß, da $\mathcal{L}'$ in eine Konfiguration $(z, \wedge, \wedge)$ gelangt war. Es soll nochmals betont werden, daß ein Zustand aus F genau dann erhalten wird, wenn der Teil des Arbeitsbandes, der das Bild des Pufferinhaltes darstellt, das Symbol λ enthält. Ist die Eingabekette noch nicht abgearbeitet, so bleibt der Leseschreibkopf vor dem ersten noch nicht gelöschten Symbol der Kette aus X' stehen. Im nächsten Schritt wird beim Vorliegen eines Zustandes (z_k, z_k) dieser in z_k geändert und der beschriebene Vorgang von neuem begonnen. Wurde jedoch ein Zustand z_j erhalten, der nicht aus F ist, so wird der Prozeß damit weitergeführt.

Wir geben nun die Konstruktion von δ aus δ' an:

Wir stellen jedem Typ von Änderungen im Pufferautomaten $\mathcal{L}'$ die Folge von Relationen aus $\mathcal{L}$ gegenüber, die das Analoge bewirken:

$$1) \delta'[z_0, x, \wedge] \supset \{(z, p, b_{20})\}$$

$$\delta'[z_0, x, \wedge] \supset \{(z, p, b_{21})\} \quad , \text{wobei } p \in \bar{P}'$$

Beide Relationen bewirken in $\mathcal{S}'$ dasselbe:

$$(z_0, xw, \wedge) \vdash_{\mathcal{S}'} (z, w, p) \quad \text{mit } w \in \Omega(X')$$

Für alle Relationen dieses Typs in $\mathcal{S}'$ muß in δ eingeführt werden:

$$\delta[z_0, x] \supset \{(z, p^{AE}, +1)\} \quad \text{falls } p \in P'$$

$$\text{bzw. } \delta[z_0, x] \supset \{((z, z), \lambda, +1)\} \quad \text{falls } p = \wedge$$

Damit gilt in $\mathcal{L}$:

$$(z_0 xw) \vdash_{\mathcal{L}} (p^{AE} zw) \quad \text{bzw.} \quad (z_0 xw) \vdash_{\mathcal{L}} (\lambda(z, z)w)$$

$$2) \delta'[z, x, p] \supset \{(z', p', b_{20})\} \quad , \text{wobei } p, p' \in P'$$

Dies bewirkt in $\mathcal{S}'$:

$$a) (z, xw, p \pi) \vdash_{\mathcal{S}'} (z', w, p \pi p') \\ \text{mit } w \in \Omega(X'), \pi \in \Omega(P') \setminus \{\wedge\}$$

$$b) (z, xw, p) \vdash_{\mathcal{S}'} (z', w, pp')$$

Für alle Relationen dieses Typs in $\mathcal{S}'$ muß in δ eingeführt werden:

$$\delta[(z, z), x] \supset \{((z, x, p, z', p', b_{20}, 1), 1, -1)\}$$

$$\delta[z, x] \supset \{((z, x, p, z', p', b_{20}, 1), 1, -1)\}$$

$$\delta[(z, x, p, z', p', b_{20}, 1), x] \supset \{((z, x, p, z', p', b_{20}, 1), x, -1)\}$$

$$\text{für } x \in X \setminus (P_A \cup P_{AE})$$

$$\delta[(z, x, p, z', p', b_{20}, 1), p^A] \supset \{((z, x, p, z', p', b_{20}, 2), p^A, +1)\}$$

$$\delta[(z, x, p, z', p', b_{20}, 1), p^{AE}] \supset \{((z, x, p, z', p', b_{20}, 3), p^{AE}, +1)\}$$

$$\delta[(z, x, p, z', p', b_{20}, 2), \bar{p}] \supset \{((z, x, p, z', p', b_{20}, 2), \bar{p}, +1)\}$$

$$\text{für } \bar{p} \in P'$$

$$\delta[(z, x, p, z', p', b_{20}, 2), p^E] \supset \{((z, x, p, z', p', b_{20}, 3), p^E, +1)\}$$

$$\delta[(z, x, p, z', p', b_{20}, 3), \bar{x}] \supset \{(z', p^E, +1)\} \quad \text{für } \bar{x} \in X$$

$$\delta[z', x] \supset \{(z', x, +1)\} \quad \text{für } x \in X \setminus X'$$

Man sieht leicht, daß damit in $\mathcal{L}$ gilt:

$$a) (p^A \pi' p'^E \bar{1} z^* x w) \vdash_{\mathcal{L}}^* (p^A \pi' p'^E \bar{1} z' w),$$

wohei $\pi = \pi' p^*$ und $\pi' \in \Omega(P'), \bar{1} \in \Omega(\{1\}), z^* \in \{z, (z, z)\}$

$$b) (p^{AE} \bar{1} z^* x w) \vdash_{\mathcal{L}}^* (p^{AE} \bar{1} z' w)$$

$$3) \delta'[z, x, p] \supset \{(z', p', b_{21})\}$$

Dies bewirkt in $\mathcal{S}'$:

$$a) (z, x w, p \pi) \vdash_{\mathcal{S}'} (z', w, \pi p')$$

$$b) (z, x w, p \bar{p}) \vdash_{\mathcal{S}'} (z', w, \bar{p} p')$$

$$c) (z, x w, p) \vdash_{\mathcal{S}'} (z', w, p')$$

Für alle Relationen dieses Typs in $\mathcal{S}'$ muß in δ eingeführt werden:

$$\delta[(z, z), x] \supset \{((z, x, p, z', p', b_{21}, 1), 1, -1)\}$$

$$\delta[z, x] \supset \{((z, x, p, z', p', b_{21}, 1), 1, -1)\}$$

$$\delta[(z, x, p, z', p', b_{21}, 1), \alpha] \supset \{((z, x, p, z', p', b_{21}, 1), \alpha, -1)\}$$

für $\alpha \in X \setminus (P_A \cup P_{AE})$

$$\delta[(z, x, p, z', p', b_{21}, 1), p^A] \supset \{((z, x, p, z', p', b_{21}, 2), \wedge, +1)\}$$

$$\delta[(z, x, p, z', p', b_{21}, 1), p^{AE}] \supset \{(z', p^{AE}, +1)\}$$

$$\delta[(z, x, p, z', p', b_{21}, 2), p^*] \supset \{((z, x, p, z', p', b_{21}, 2), p^*, +1)\}$$

für $p^* \in X \setminus P_E$

$$\delta[(z, x, p, z', p', b_{21}, 2), \bar{p}^E] \supset \{((z, x, p, z', p', b_{21}, 3), \bar{p}, +1)\}$$

$$\delta[(z, x, p, z', p', b_{21}, 3), \alpha] \supset \{(z', p'^E, +1)\} \text{ für } \alpha \in X$$

$$\delta[z', \alpha] \supset \{(z', \alpha, +1)\} \text{ für } \alpha \in X \setminus X'$$

Man sieht leicht, daß entsprechend der Fallunterscheidung für $\mathcal{S}'$ in $\mathcal{L}$ gilt:

$$a) (p^A \bar{p} \pi'' p'^E \bar{1} z^* x w) \vdash_{\mathcal{L}}^* (\bar{p}^A \pi'' p'' p'^E \bar{1} z' w) \text{ für } \pi = \bar{p} \pi'' p''$$

$$b) (p^A \bar{p}^E \bar{1} z^* x w) \vdash_{\mathcal{L}}^* (\bar{p}^A p'^E \bar{1} z' w)$$

$$c) (p^{AE} \bar{1} z^* x w) \vdash_{\mathcal{L}}^* (p^{AE} \bar{1} z' w)$$

$$4) (0) \delta'[z, x, \wedge] \supset \{(z', p', b_{20})\}$$

$$(1) \delta'[z, x, \wedge] \supset \{(z', p', b_{21})\}$$

Beide Relationen bewirken in $\mathcal{S}'$ dasselbe:

$$(z, x w, \wedge) \vdash_{\mathcal{S}'} (z', w, p')$$

Für alle Relationen dieses Typs in $\mathcal{L}'$ muß in δ eingeführt werden:

$$\begin{aligned} \delta[(z, z), x] &\supset \{((z, x, \wedge, z', p', b_{21}, 1), 1, -1)\}^1) \\ \delta[z, x] &\supset \{((z, x, \wedge, z', p', b_{21}, 1), 1, -1)\}^1) \\ \delta[(z, x, \wedge, z', p', b_{21}, 1), x] &\supset \{((z, x, \wedge, z', p', b_{21}, 1), x, -1)\}^1) \\ &\quad \text{für } x \in X \setminus (P_A \cup P_E \cup P_{AE}) \\ \delta[(z, x, \wedge, z', p', b_{21}, 1), \lambda] &\supset \{(z', p', AE, +1)\}^1) \\ \delta[z', x] &\supset \{(z', x, +1)\} \quad \text{für } x \in X \setminus X' \end{aligned}$$

Man sieht leicht, daß damit in $\mathcal{L}$ gilt:

$$(\lambda \bar{1} z^* x w) \vdash_{\mathcal{L}}^* (p', AE \bar{1} z' w) \quad , \text{wobei } \bar{1}' \in \Omega(\{1\})$$

$$5) \delta'[z, x, p] \supset \{(z', \wedge, b_{20})\}$$

Diese Relation bewirkt in $\mathcal{L}'$:

$$a) (z, x w, p \pi) \vdash_{\mathcal{L}'} (z', w, p \pi)$$

$$b) (z, x w, p) \vdash_{\mathcal{L}'} (z', w, p)$$

Für alle Relationen dieses Typs in $\mathcal{L}'$ muß in δ eingeführt werden:

$$\begin{aligned} \delta[(z, z), x] &\supset \{((z, x, p, z', \wedge, b_{20}, 1), 1, -1)\} \\ \delta[z, x] &\supset \{((z, x, p, z', \wedge, b_{20}, 1), 1, -1)\} \\ \delta[(z, x, p, z', \wedge, b_{20}, 1), x] &\supset \{((z, x, p, z', \wedge, b_{20}, 1), x, -1)\} \\ &\quad \text{für } x \in X \setminus (P_A \cup P_{AE}) \\ \delta[(z, x, p, z', \wedge, b_{20}, 1), p^A] &\supset \{(z', p^A, +1)\} \\ \delta[(z, x, p, z', \wedge, b_{20}, 1), p^{AE}] &\supset \{(z', p^{AE}, +1)\} \\ \delta[z', x] &\supset \{(z', x, +1)\} \quad \text{für } x \in X \setminus X' \end{aligned}$$

Man sieht leicht, daß damit in $\mathcal{L}$ gilt:

$$a) (p^A \pi' \bar{p}^E \bar{1} z x w) \vdash_{\mathcal{L}}^* (p^A \pi' \bar{p}^E \bar{1}' z' w) \quad \text{mit } \pi = \pi' \bar{\pi}$$

$$b) (p^{AE} \bar{1} z x w) \vdash_{\mathcal{L}}^* (p^{AE} \bar{1}' z' w)$$

$$6) \delta'[z, x, p] \supset \{(z', \wedge, b_{21})\}$$

¹⁾ $i=0$,falls 4(0) vorlag
 $i=1$,falls 4(1) vorlag

Diese Relation bewirkt in $\mathcal{X}'$:

$$a) \pi = \bar{p} \pi' \bar{p}' : (z, xw, p \bar{p} \pi' \bar{p}') \vdash_{\mathcal{X}'} (z', w, \bar{p} \pi' \bar{p}')$$

$$b) \pi = \bar{p} : (z, xw, p \bar{p}) \vdash_{\mathcal{X}'} (z', w, \bar{p})$$

$$c) (z, xw, p) \vdash_{\mathcal{X}'} (z', w, \wedge)$$

Für alle Relationen dieses Typs in $\mathcal{X}'$ muß in δ eingeführt werden:

$$\delta[(z, z), x] \supset \{((z, x, p, z', \wedge, b_{21}, 1), 1, -1)\}$$

$$\delta[z, x] \supset \{((z, x, p, z', \wedge, b_{21}, 1), 1, -1)\}$$

$$\delta[(z, x, p, z', \wedge, b_{21}, 1), \alpha] \supset \{((z, x, p, z', \wedge, b_{21}, 1), \alpha, -1)\}$$

$$\text{für } \alpha \in X \setminus (P_A \cup P_{AE})$$

$$\delta[(z, x, p, z', \wedge, b_{21}, 1), p^A] \supset \{((z, x, p, z', \wedge, b_{21}, 2), \wedge, +1)\}$$

$$\delta[(z, x, p, z', \wedge, b_{21}, 1), p^{AE}] \supset \{((z', z'), \lambda, +1)\}$$

$$\delta[(z, x, p, z', \wedge, b_{21}, 2), \bar{p}^E] \supset \{(z', \bar{p}^{AE}, +1)\}$$

$$\delta[(z, x, p, z', \wedge, b_{21}, 2), p^*] \supset \{(z', p^{*A}, +1)\}$$

$$\delta[z', \alpha] \supset \{(z', \alpha, +1)\} \quad \text{für } \alpha \in X \setminus X'$$

$$\delta[(z', z'), \alpha] \supset \{((z', z'), \alpha, +1)\} \quad \text{für } \alpha \in X \setminus X'$$

Man sieht leicht, daß damit in $\mathcal{X}$ gilt:

$$a) \pi = \bar{p} \pi' \bar{p}' : (p \bar{p} \pi' \bar{p}', {}^E \bar{1} z x w) \vdash_{\mathcal{X}}^* (\bar{p}^A \pi' \bar{p}', {}^E \bar{1}' z' w)$$

$$b) \pi = \bar{p} : (p \bar{p}^E \bar{1} z x w) \vdash_{\mathcal{X}}^* (\bar{p}^{AE} \bar{1}' z' w)$$

$$c) (p^{AE} \bar{1} z x w) \vdash_{\mathcal{X}}^* (\lambda \bar{1}' (z', z') w)$$

$$7)(0) \delta'[z, x, \wedge] \supset \{(z', \wedge, b_{20})\}$$

$$(1) \delta'[z, x, \wedge] \supset \{(z', \wedge, b_{21})\}$$

Beide Relationen bewirken dasselbe:

$$(z, xw, \wedge) \vdash_{\mathcal{X}'} (z', w, \wedge)$$

Für alle Relationen dieses Typs muß in δ eingeführt werden:

$$\delta[(z, z), x] \supset \{((z, x, \wedge, z', \wedge, b_{21}, 1), 1, -1)\} \quad 1)$$

$$\delta[z, x] \supset \{((z, x, \wedge, z', \wedge, b_{21}, 1), 1, -1)\} \quad 1)$$

1) i=0, falls 7(0) vorlag; i=1, falls 7(1) vorlag

$$\delta[(z, x, \wedge, z', \wedge, b_{21}, 1), \alpha] \supset \{((z, x, \wedge, z', \wedge, b_{21}, 1), \alpha, -1)\}^{1)}$$

für $\alpha \in X \setminus (P_A \cup P_E \cup P_{AE})$

$$\delta[(z, x, \wedge, z', \wedge, b_{21}, 1), \lambda] \supset \{((z', z'), \lambda, +1)\}^{1)}$$

$$\delta[(z', z'), \alpha] \supset \{((z', z'), \alpha, +1)\} \text{ für } \alpha \in X \setminus X'$$

Dann gilt in $\mathcal{Z}$:

$$(\lambda \bar{1}zxw) \vdash_x^h (\lambda \bar{1}'(z', z')w).$$

Aufgrund der Konstruktion gilt nun:

Gibt es in $\mathcal{Z}'$ eine Übergangsfolge, die $(z_0, w, \wedge)$ mit $w \neq \wedge$ überführt in $(z, \wedge, \wedge)$, so gibt es in $\mathcal{Z}$ eine Übergangsfolge, die (z_0, w) überführt in $(u(z, z))$ mit $(z, z) \in F$ und umgekehrt, d.h., es gilt $w \in T(\mathcal{Z}') \Leftrightarrow w \in T(\mathcal{Z})$.

Damit gilt $T(\mathcal{Z}') = T(\mathcal{Z}) \setminus \{\wedge\}$ und so auch $T(\mathcal{Z}) = T(\mathcal{Z}') \cup \{\wedge\}$.

Über den Zusammenhang zwischen linear-beschränkten Automaten und kontextsensitiven Sprachen gilt der folgende Satz (s.z.B. GINSBURG/ROSE (1966)).

Satz: Eine Menge L ist genau dann eine kontext-sensitive Sprache, wenn es einen linear-beschränkten Automaten $\mathcal{Z}$ gibt, so daß gilt: $L = T(\mathcal{Z})$.

In Verbindung mit dem obigen Lemma ergibt sich damit unmittelbar der Satz:

Satz 4-3: Jede Puffersprache ist eine kontext-sensitive Sprache.

Wir haben damit die Puffersprachen nach zwei Seiten abgegrenzt. Einmal zeigten wir, daß jede reguläre Sprache eine deterministische Puffersprache ist, und zum andern ist jede nichtdeterministische Puffersprache auch eine kontext-sensitive Sprache. Es gibt also keine Puffersprachen, die Typ-0-, aber keine Typ-1-

¹⁾ $i=0$, falls $7(0)$ vorlag
 $i=1$, falls $7(1)$ vorlag

(kontext-sensitive) Sprachen sind.

Für die deterministischen Puffersprachen soll nun noch bewiesen werden, daß sie nicht mit einem der Typen 1 oder 2 zusammenfallen.

Satz 4-4: (a) Es gibt kontextfreie, nicht-reguläre Sprachen, die deterministische Puffersprachen sind.

(b) Es gibt kontextfreie, nicht-reguläre Sprachen, die keine deterministischen Puffersprachen sind.

Bew.: (a) Die Menge $L := \{a^n b a^n \mid n \in \mathbb{N}\}$, für die wir im Beweis von Satz 4-2 (b) einen einfachen deterministischen Pufferautomaten angegeben, der sie annimmt, stellt ein Beispiel einer kontextfreien, nicht regulären Sprache dar.

(b) Analog zu dem Vorgehen im Beweis von Satz 4-1 (b) zeigt man, daß es keinen einfachen deterministischen Pufferautomaten $\mathcal{M}$ gibt, der die kontextfreie, nicht-reguläre Sprache $L := \{w w^R \mid w \in \Omega(\{a, b\}^*) \setminus \{\Lambda\}\}$, wobei w^R wie üblich die Spiegelung des Wortes w bezeichne, annimmt. Aus der entgegengesetzten Annahme ergibt sich der Widerspruch, daß es auch Ketten der Form $a^2 i j b b a^2 i$ mit $i \neq j$ gibt, die $\in T(\mathcal{M})$ sind.

Satz 4-5: (a) Es gibt kontextsensitive, nicht-kontextfreie Sprachen, die deterministische Puffersprachen sind.

(b) Es gibt kontextsensitive, nicht-kontextfreie Sprachen, die keine deterministische Puffersprachen sind.

Bew.: (a) Wir geben einen einfachen deterministischen Pufferautomaten $\mathcal{M}$ an, der die kontextsensitive, nicht-kon-

textfreie Sprache $L := \{a^n b^n a^n \mid n \in \mathbb{N}\}$ annimmt.

$\mathcal{L}_D = (\{a, b\}, \{z_0, z_1, z_2, z_3\}, \{a, b\}, \{b_{20}, b_{21}\}, d, z_0)$,

wobei d folgendermaßen gegeben ist:

$$d(z_0, a, \Lambda) = (z_0, a, b_{20})$$

$$d(z_0, a, a) = (z_0, a, b_{20})$$

$$d(z_0, a, b) = (z_3, \Lambda, b_{20})$$

$$d(z_0, b, \Lambda) = (z_3, b, b_{20})$$

$$d(z_0, b, a) = (z_1, b, b_{21})$$

$$d(z_0, b, b) = (z_3, \Lambda, b_{20})$$

$$d(z_1, a, \Lambda) = (z_3, a, b_{20})$$

$$d(z_1, a, a) = (z_3, \Lambda, b_{20})$$

$$d(z_1, a, b) = (z_2, \Lambda, b_{21})$$

$$d(z_1, b, \Lambda) = (z_3, b, b_{20})$$

$$d(z_1, b, a) = (z_1, b, b_{21})$$

$$d(z_1, b, b) = (z_3, \Lambda, b_{20})$$

$$d(z_2, a, \Lambda) = (z_3, a, b_{20})$$

$$d(z_2, a, a) = (z_3, \Lambda, b_{20})$$

$$d(z_2, a, b) = (z_2, \Lambda, b_{21})$$

$$d(z_2, b, \Lambda) = (z_3, b, b_{20})$$

$$d(z_2, b, a) = (z_3, \Lambda, b_{20})$$

$$d(z_2, b, b) = (z_3, \Lambda, b_{20})$$

$$d(z_3, a, \Lambda) = (z_3, a, b_{20})$$

$$d(z_3, a, a) = (z_3, \Lambda, b_{20})$$

$$d(z_3, a, b) = (z_3, \Lambda, b_{20})$$

$$d(z_3, b, \Lambda) = (z_3, b, b_{20})$$

$$d(z_3, b, a) = (z_3, \Lambda, b_{20})$$

$$d(z_3, b, b) = (z_3, \Lambda, b_{20})$$

Man überzeugt sich leicht, daß $T(\mathcal{L}_D) = L$.

(b) Die Menge $\{ww \mid w \in \Omega(\{a, b\}) \setminus \{\Lambda\}\}$ ist ein Beispiel für

eine kontextsensitive, nicht-kontextfreie Sprache.

Im Beweis von Satz 4-1 (b) zeigten wir, daß es keinen einfachen deterministischen Pufferautomaten gibt, der diese Sprache annimmt.

Damit ist gezeigt, daß die deterministischen Puffersprachen mit keinem der vier Sprachtypen von CHOMSKY identifizierbar sind.

An dem Beispiel der kontextfreien Sprache $\{b^{1+k}a^1 \mid k, l \in \mathbb{N}\}$ wird im Beweis von Satz 5-4 ein anderer Beweis für Satz 4-4(b) gegeben.

Es gilt sogar der Satz, daß es deterministische kontextfreie Sprachen gibt, die keine deterministische Puffersprachen sind. Der Beweis läßt sich am Beispiel der Sprache $\{wvcv^R_w \mid w \in \Omega(\{a, b\}) \setminus \{\wedge\} \wedge v \in \Omega(\{0, 1\}) \setminus \{\wedge\}\}$ mit denselben Worten wie der Beweis des späteren Satzes 5-1 führen.

In dem genannten Beweis wird auch noch an einem anderen Beispiel Satz 4-5(b) bewiesen.

5. Operationen mit deterministischen Puffersprachen

In diesem Abschnitt werden wir das Verhalten deterministischer Puffersprachen bei Anwendung einiger charakteristischer Operationen betrachten. Die Ergebnisse sind fast durchweg negativ.

Der bequemeren Sprechweise halber wollen wir den Begriff des Pufferbildes einführen.

Def.: Gilt in einem deterministischen Pufferautomaten

$$\begin{aligned} \mathcal{A}_D &= (X, Z, P, B_2, d, z_0) \\ (z_0, u_1 u_2, \Lambda) &\vdash_{\mathcal{A}_D}^* (z, u_2, \pi^1 \pi^2) \vdash_{\mathcal{A}_D}^* (z', u_2, \pi^2 \pi) \\ \text{mit } z, z' &\in Z, u_1, u_2, u \in \Omega(X), \pi^1, \pi^2, \pi \in \Omega(P), \text{ so hei\ss t} \\ \pi &\text{ das } \underline{\text{Pufferbild von } u} \text{ (in } \mathcal{A}_D \text{).} \end{aligned}$$

Satz 5-1: Die Klasse der deterministischen Puffersprachen ist bzgl. der Durchschnittsbildung nicht abgeschlossen.

Bew.: Der Beweis erfolgt am Beispiel der Sprache

$$\begin{aligned} L &= \{wvcvw / w \in \Omega(\{a,b\}) \setminus \{\Lambda\} \wedge v \in \Omega(\{0,1\}) \setminus \{\Lambda\}\}, \\ \text{von der wir zeigen werden, da\ss sie keine deterministische} \\ \text{Puffersprache ist und die als Durchschnitt der beiden} \\ \text{deterministischen Puffersprachen } L_1 &= \{wv_1cv_2w \\ / w \in \Omega(\{a,b\}) \setminus \{\Lambda\} \wedge v_1, v_2 &\in \Omega(\{0,1\}) \setminus \{\Lambda\}\} \text{ und} \\ L_2 &= \{w_1vcvw_2 / w_1, w_2 \in \Omega(\{a,b\}) \setminus \{\Lambda\} \wedge v \in \Omega(\{0,1\})\} \\ &\text{gebildet werden kann.} \end{aligned}$$

Da\ss L_1 und L_2 deterministische Puffersprachen sind, zeigt man leicht durch Konstruktion von deterministischen Pufferautomaten nach folgendem Prinzip: Um W\u00f6rter aus L_1 anzunehmen, wird zun\u00e4chst das Pufferbild der ersten Teilkette w in den Puffer geschrieben, ohne \u00c4nderung des Puffers wird nachgepr\u00fcft, ob v_1 und v_2 Ele-

mente aus $\Omega(\{0,1\})$ sind und anschließend wird die zweite Teilkette w mit dem Pufferinhalt verglichen. Ein Automat, der L_2 annimmt, schreibt zunächst ein Sondersymbol in den Puffer, überprüft, ob $w_1 \in \Omega(\{a,b\})$, löscht beim Einlesen von v das Sondersymbol und schreibt gleichzeitig das Pufferbild von v ein, vergleicht es beim Einlesen der zweiten Teilkette v damit und schreibt beim Leerwerden des Puffers wiederum ein Sondersymbol ein, das beim Einlesen des ersten Symbols von w_2 gelöscht wird. Dann wird nur noch einmal etwas in den Puffer geschrieben, wenn $w_2 \notin \Omega(\{a,b\})$.

Wir wollen nun zeigen, daß L keine deterministische Puffersprache ist, d.h., daß es keinen deterministischen Pufferautomaten $\mathcal{L}_D$ gibt, so daß gilt: $T(\mathcal{L}_D) = L$. L ist keine reguläre Sprache, kann also nicht von einem endlichen erkennenden Automaten angenommen werden.

D.h., um L anzunehmen, muß von der Möglichkeit des Speicherns von Information auf einem Speicherband Gebrauch gemacht werden. Nehmen wir an, es gäbe einen deterministischen Pufferautomaten $\mathcal{L}_D = (X, Z, P, B_2, d, z_0)$, der L annimmt.

Wir wollen zunächst zeigen, daß sich nach Abarbeiten der "Hälfte" eines Wortes aus L der Automat in einer durch den Zustand und den Pufferinhalt durch wv umkehrbar eindeutig bestimmten Konfiguration befinden muß: Würde nämlich gelten

$$(z_0, wvcvw, \Lambda) \vdash_{\mathcal{L}_D}^* (z, cvw, \pi)$$

und $(z_0, w'v'cv'w', \Lambda) \vdash_{\mathcal{L}_D}^* (z, cv'w', \pi)$

mit $z \in Z$, $\pi \in \Omega(P)$ und $w' \neq w \vee v' \neq v$, so würde wegen der

Determiniertheit des Automaten mit $wvcvw$ auch $w'v'cvw$ angenommen, und dieses Wort ist nicht in L .

Die Anzahl der Elemente von Z sei α_1 und die von P sei α_2 (≥ 2). Mit Ketten im Puffer der Länge m , lassen sich mithilfe der Zustände α_1^{m+1} verschiedene Konfigurationen bilden. Andererseits gibt es 2^{1+k} verschiedene Ketten wv mit $|w| = 1$ und $|v| = k$. Sollen Ketten dieser Länge umkehrbar eindeutig Konfigurationen im Pufferautomaten zugeordnet werden, so muß für die Pufferlänge m gelten: $2^{1+k} < \alpha_1^{m+1}$

D.f. $(*) \quad 1 \cdot c_1 + k \cdot c_1 < (m+1) \cdot c_2 + c_3$

mit Konstanten $c_1, c_2 > 0, c_3 \geq 0$.

Für ein Wort aus L existiert eine Übergangsfolge

$$(z_0, wvcvw, \wedge) \vdash_{\mathcal{A}_D}^* (z^1, vcvw, \pi^1) \vdash_{\mathcal{A}_D}^* (z^2, cvw, \pi^2) \vdash_{\mathcal{A}_D}^* (z^3, w, \pi^3) \quad \text{mit } z^1, z^2, z^3 \in Z, \pi^1, \pi^2, \pi^3 \in \Omega(P).$$

Da die Pufferlänge linear mit der Länge von w wächst, gibt es in L beliebig viele Wörter, für die gilt

$|\pi^1| > |vcv|$, wobei $|v|$ - in Abhängigkeit von π^1 - beliebig groß werden kann. D.h., nach dem Einlesen von w enthält der Puffer so viele Symbole, daß sie im Verlauf des Einlesens von vcv nicht alle gelöscht werden können. - Beim Einlesen eines Eingabesymbols kann ja höchstens ein Symbol aus dem Puffer ausgelesen werden. -

Um nun die beiden Ketten v - sie seien als $v^{(1)}$ und $v^{(2)}$ bezeichnet - auf Übereinstimmung zu prüfen, ist es nötig, nach dem vollständigen Auslesen des Pufferbildes von w ihre Pufferbilder zu vergleichen. Da das Pufferbild der ersten Kette $v^{(1)}$ während des Einlesens der zweiten Kette $v^{(2)}$ für den Pufferlesekopf nicht erreichbar war, wer-

den die Pufferbilder von $v^{(1)}$ und von $v^{(2)}$ unabhängig voneinander in den Puffer geschrieben. Aus (*) folgt, daß es Wörter in L gibt, für die gilt: Zum Zeitpunkt des vollständigen Ausgelesenseins des Pufferbildes von w ist die Anzahl K der Symbole im Puffer, d.h. die Länge des Pufferbildes von wcv , eine lineare Funktion von $|v| = k$. Da der Pufferlesekopf lediglich Zugriff zu einem am linken Ende des Puffers befindlichen Symbol hat, müssen die Pufferbilder von $v^{(1)}$ und von $v^{(2)}$ abwechselnd an das linke Ende geschoben werden. Wegen der endlichen Zustandszahl können aber während eines solchen Vorganges nur eine endliche Anzahl von Puffersymbolen verglichen und damit auch gelöscht werden. Der Vergleich beansprucht deshalb eine Anzahl von Zustandsübergängen, -wobei jeweils auch ein Symbol vom Eingabeband eingelesen wird-, die proportional zu K^2 und damit auch zu k^2 ist. Da aber nach dem Einlesen der Teilkette $wvcv$ nur noch $|w| = 1$ Zustandsübergänge möglich sind, gibt es nach (*) Wörter in L , für die der Vergleich nicht durchführbar ist, so daß auch Wörter der Form $wvcv'w$ mit $v \neq v'$ angenommen werden. Damit haben wir einen Widerspruch zur Annahme, daß L eine deterministische Puffersprache sei.

Satz 5-2: Die Klasse der deterministischen Puffersprachen ist bzgl. der Vereinigung nicht abgeschlossen.

Bew.: Es wird gezeigt, daß die als Vereinigung zweier deterministischer Puffersprachen L_1 und L_2 gebildete Sprache (nach RABIN (1963)) $L = \{uvdu\} \cup \{uvev\}$, wobei $u \in \Omega(\{a,b\}) \setminus \{\Lambda\}$, $v \in \Omega(\{0,1\}) \setminus \{\Lambda\}$ keine deterministische Puffersprache darstellt.

Man überzeugt sich leicht, daß L_1 und L_2 deterministische Puffersprachen sind: Ein deterministischer Pufferautomat, der L_1 annimmt, kann dergestalt arbeiten, daß er das Pufferbild der Teilkette u in den Puffer schreibt, allein mithilfe von Zustandsübergängen nachprüft, ob die bis d folgende Teilkette ein Element aus $\Omega(\{0,1\})$ darstellt und nach dem Einlesen von d die folgenden Symbole mit dem Pufferinhalt, der dabei gleichzeitig gelöscht wird, vergleicht. Um L_2 anzunehmen, verwendet man einen deterministischen Pufferautomaten, der zunächst ein Sonder-symbol in den Puffer schreibt, ohne weitere Benutzung des Puffers feststellt, ob $u \in \Omega(\{a,b\})$, beim Einlesen von v das einzige im Puffer stehende Symbol löscht und gleichzeitig das Pufferbild von v einschreibt, das dann nach Einlesen von e mit der folgenden Teilkette v verglichen wird.

Nehmen wir nun an, es gebe einen deterministischen Pufferautomaten $\mathcal{L}_D = (X, Z, P, B_2, d, z_0)$, der L annimmt.

Wie im Beweis von Satz 5-1 zeigt man, daß sich der Automat nach Einlesen einer Teilkette uv in einer durch den Zustand und den Pufferinhalt umkehrbar eindeutig durch uv bestimmten Konfiguration befinden muß.

Außerdem gilt auch hier wieder das im Beweis von Satz 5-1 über die Pufferlänge Gesagte. Daraus folgt dann, daß es in L beliebig viele Wörter gibt (mit $|u| \gg |v|$), für die die Länge des Pufferbildes von u größer als die Länge von v ist, d.h. es gibt in $\mathcal{L}_D$ eine Übergangsfolge $(z_0, uvev, \wedge) \vdash_{\mathcal{L}_D}^* (z, ev, \pi)$ mit $|\pi| > |ev|$.

Während des Abarbeitens von ev kann aber der Puffer nicht

leer werden und damit das Wort $uvuv$ nicht angenommen werden, obwohl es Element aus L ist.

Damit erhalten wir einen Widerspruch zur Annahme, daß es einen deterministischen Pufferautomaten gebe, der L annimmt.

Def.: Seien X_1 und X_2 nichtleere Mengen. Ein Homomorphismus τ von $\Omega(X_1)$ in $\Omega(X_2)$ ist eine Abbildung von $\Omega(X_1)$ in $\Omega(X_2)$, so daß gilt:

$$\tau(\wedge) = \wedge$$

$$\text{und } \tau(x^1 \dots x^k) = \tau(x^1) \dots \tau(x^k)$$

$$\text{für alle } k \geq 1 \text{ und } x^i \in X_1 \quad (1 \leq i \leq k).$$

Satz 5-3: Die Klasse der deterministischen Puffersprachen ist bezüglich der Homomorphismenbildung nicht abgeschlossen.

Bew.: Zum Beweis betrachten wir die Sprache $L = \{fuvdu\} \cup \{guvev\}$, wobei $u \in \Omega(\{a, b\}) \setminus \{\wedge\}$ und $v \in \Omega(\{0, 1\}) \setminus \{\wedge\}$. Man überzeugt sich leicht, daß L eine deterministische Puffersprache darstellt.

Sei nun τ der folgendermaßen definierte Homomorphismus:

$$\tau(f) = \tau(g) = \wedge; \tau(a) = a; \tau(b) = b; \tau(0) = 0;$$

$$\tau(1) = 1; \tau(d) = d; \tau(e) = e$$

Dann ist $\tau(L) = \{uvdu\} \cup \{uvev\}$, und dies ist nach dem Beweis von Satz 5-2 keine deterministische Puffersprache.

Satz 5-4 : Die Klasse der deterministischen Puffersprachen ist bezüglich der Spiegelung nicht abgeschlossen.

Bew.: Am Beispiel der Sprache $L = \{a^k b^{k+1} \mid k, l \in \mathbb{N}\}$ soll die Behauptung gezeigt werden.

Ein deterministischer Pufferautomat, der L annimmt, kann folgende Vorgehensweise haben: Im Automaten wird während des Einlesens von a^k ein umkehrbar eindeutiges Bild davon erzeugt, z.B. durch Einschreiben einer Kette der Länge k in den Puffer. Während des Einlesens von b 's wird diese Kette Symbol für Symbol getilgt. Beim Leerwerden des Puffers wird nochmals ein Symbol eingeschrieben, um zu erreichen, daß die Kette der b 's um mindestens ein Symbol länger ist, als die der a 's, und beim Einlesen eines weiteren b wird das Symbol gelöscht.

Die Spiegelung der Sprache L , die Sprache $L^R = \{b^{1+k}a^k / k, 1 \in \mathbb{N}\}$ ist keine deterministische Puffersprache,

Ein deterministischer Pufferautomat, der L annehmen sollte, müßte nämlich in eine durch den Zustand und den Pufferinhalt gekennzeichnete durch b^k umkehrbar eindeutig bestimmte Konfiguration gelangen. Wegen der Determiniertheit des Automaten wird dann aber, da 1 beliebig groß werden kann, der Pufferinhalt so groß, daß er während des Einlesens von a^k nicht mehr gelöscht werden kann.

Satz 5-5 : Die Klasse der deterministischen Puffersprachen ist bezüglich der Konkatenation nicht abgeschlossen.

Der Beweis ist ähnlich wie in Satz 5-4. Hier wird die Konkatenation der deterministischen Puffersprachen $L_1 = \{b^1 / 1 \in \mathbb{N}\}$ und $L_2 = \{b^k a^k / k \in \mathbb{N}\}$ betrachtet, die die Sprache $L_1 L_2 = \{b^{1+k} a^k / k, 1 \in \mathbb{N}\}$ ergibt, und diese Sprache ist keine deterministische Puffersprache.

Satz 5-6 : Die Klasse der deterministischen Puffersprachen ist bezüglich der Differenzbildung nicht abgeschlossen.

Der Beweis erfolgt am Beispiel der Sprache $L = \{a^{k+1}b^k / k, l \in \mathbb{N}\}$, die nach dem Beweis von Satz 5-4 keine deterministische Puffersprache ist und die die Differenz $L = L_1 \setminus L_2$ der deterministischen Puffersprachen $L_1 = \{a^i b^j / i, j \in \mathbb{N}\}$ und $L_2 = \{a^i b^{i+m} / i \in \mathbb{N}, m \in \mathbb{N} \setminus \{0\}\}$ darstellt.

Zwei weitere negative Resultate betreffen die Konkatenation von deterministischen Puffersprachen und regulären Sprachen.

Satz 5-7: a) Es gibt eine deterministische Puffersprache L und eine reguläre Sprache R , so daß RL keine deterministische Puffersprache ist.

b) Es gibt eine deterministische Puffersprache L' und eine reguläre Sprache R' , so daß $L'R'$ keine deterministische Puffersprache ist.

Bew.: a) s. Beweis von Satz 5-5

b) Zum Beweis betrachten wir die Sprache L_1 , die aus solchen Wörtern aus $\Omega(\{a, b\}) \setminus \{\Lambda\}$ besteht, die die gleiche Anzahl von a's und b's enthalten und die reguläre Sprache $L_2 = \{a^i / i \in \mathbb{N}\}$. Man überlegt sich leicht, daß L_1 eine deterministische Puffersprache darstellt.

Nehmen wir nun an, es gebe einen deterministischen Pufferautomaten $\mathcal{A}_D = (X, Z, P, B_2, d, z_0)$, für den gilt:

$$T(\mathcal{A}_D) = L_1 L_2.$$

Wie oben sieht man, daß es Wörter $b^i a^i a^k b^k \in L_1$ ($i, k \in \mathbb{N}$) mit genügend großem k gibt, so daß sich der Automat nach dem Einlesen von $b^i a^i a^k$ in einer durch Zustand und Pufferinhalt umkehrbar eindeutig durch a^k bestimmten Konfiguration befinden muß, d.h., es muß gelten

$$(z_0, b^i a^i a^k b^k, \Lambda) \vdash_{\mathcal{A}_D}^* (z, a^{k-1} b^k, \Lambda), \text{ da } b^i a^{i+1} \text{ auch } \in L_1 L_2$$

und $(z, a^{k-1} b^k, \Lambda) \vdash_{\mathcal{A}_D}^* (z', b^k, \pi)$ mit $\pi \in \Omega(P) \setminus \{\Lambda\}$.

Wegen der Determiniertheit von $\mathcal{L}_D$ gilt dann auch

$$(z_0, b^i a^i a^k, \wedge) \vdash_{\mathcal{L}_D}^* (z', \wedge, \pi).$$

D.h. $b^i a^i a^k \notin T(\mathcal{L}_D)$, was wegen $b^i a^i a^k \in L_1 L_2$ einen Widerspruch zur Annahme, daß $T(\mathcal{L}_D) = L_1 L_2$ sein soll, darstellt.

Als nächstes wollen wir noch eine Operation einführen, bzgl. der die Klasse der deterministischen Puffersprachen abgeschlossen bleibt.

Def.: Sei L deterministische Puffersprache. Dann wird die Minimisation $\min L$ definiert (s. ROSENBERG (1967)) als

$$\min L = \{y \mid y \in L \wedge (x < y \Rightarrow x \notin L)\}, \text{ wobei } x < y \text{ bedeutet,}$$

daß eine nichtleere Kette z existiert, so daß $y = xz$ gilt.

Dann gilt der Satz:

Satz 5-8: Die Klasse der deterministischen Puffersprachen
ist bezüglich der Minimisation abgeschlossen.

Bew.: Analog zu einem entsprechenden Beweis in ROSENBERG geben wir an, wie zu jedem deterministischen Pufferautomaten $\mathcal{L}_D = (X, Z, P, B_2, d, z_0)$ ein deterministischer Pufferautomat $\mathcal{L}'_D = (X', Z', P', B_2, d', z'_0)$ konstruiert werden kann, der gerade die Sprache $\min T(\mathcal{L}_D)$ annimmt.

$\mathcal{L}'_D$ arbeitet so lange wie $\mathcal{L}_D$, bis eine Teilkette des Eingabewortes akzeptiert wird und geht dann in eine Konfiguration über, die nicht mehr zu einem leeren Puffer führen kann:

$\bar{p}$ sei ein ausgezeichnetes Element von P . $X' := X$,
 $Z' := Z \cup \{z', z''\}$ mit $z', z'' \notin Z$, $P' := P$, $z'_0 := z'$ und für d' gilt:

$$d'(z, x, p) = \begin{cases} (\bar{z}, p', b_{21}), & \text{falls } p \neq \wedge \text{ und } d(z, x, p) = (\bar{z}, p, b_{21}) \\ (z'', \bar{p}, b_{20}), & \text{falls } p = \wedge \end{cases}$$

Für alle $(z', x, p) \in \{z'\} \times X' \times (P' \cup \{\Lambda\})$ wird eingefügt:

$$d'(z', x, p) = (\bar{z}, p', b_{21}) \text{ , falls } d(z_0, x, p) = (\bar{z}, p', b_{21})$$

und für alle $(z'', x, \bar{p}) \in \{z''\} \times X' \times \{\bar{p}\}$ wird eingefügt

$$d'(z'', x, \bar{p}) = (z'', \bar{p}, b_{20}) .$$

Man überlegt sich leicht, daß aufgrund dieser Konstruktion gilt: $T(\mathcal{X}'_D) = \min T(\mathcal{X}_D)$.

6. Pufferautomaten und kombinatorische Systeme

Bisher betrachteten wir eine Puffersprache als die Menge der von einem einfachen Pufferautomaten angenommenen Wörter. Neben dieser in der Automatentheorie gebräuchlichen Vorgehensweise ist es üblich, zugleich eine Darstellung einer Sprache als eine von einer Grammatik oder allgemeiner einem kombinatorischen System erzeugten Menge zu fordern.

Wir werden in diesem Abschnitt spezielle kombinatorische Systeme einführen, deren Erzeugungskapazität -ungenau gesprochen- der Annahmekapazität der einfachen Pufferautomaten äquivalent ist.

In Anlehnung an DAVIS (1958) definieren wir:

Def.: Seien t, u, v, t', u', v' sechs nicht notwendig verschiedene, möglicherweise leere Wörter über einer endlichen, nicht-leeren Menge. Das zweistellige Wortprädikat $\mathcal{R}_{t,u,v}^{t',u',v'}(X, Y)$, das für gegebene Wörter X_0, Y_0 dann und nur dann wahr ist, wenn (möglicherweise leere) Wörter Q_0, R_0 existieren, so daß $X_0 = tQ_0uR_0v$ und $Y_0 = t'Q_0u'R_0v'$, wird die t, u, v, t', u', v' zugeordnete Produktion genannt und als $(tQuRv, t'Qu'Rv')$ geschrieben.

Ein kombinatorisches System ist gegeben durch ein nicht-leeres Wort und eine endliche Menge von Produktionen.

Die folgenden Definitionen, die eine Spezialisierung der kombinatorischen Systeme und das Hantieren mit ihnen zum Inhalt haben, werden sich in ihrer Formulierung möglichst eng an die in der Mathematischen Linguistik gebräuchliche Begriffsbildung anschließen (s.z.B. SCHNEIDER (1965)).

Def.: Ein B-System ist ein geordnetes Quintupel $B=(V,V',T,\Pi,S)$, wobei gilt:

(a) V, V' und T sind endliche, nichtleere Mengen mit

$$V' \cup T \subseteq V \quad \text{und} \quad V' \cap T = \emptyset$$

(b) $S \in V'$

(c1) Π ist eine endliche, nichtleere Menge von Produktionen der Form

$$(1) \quad (QzR, Qxz'R)$$

$$(2) \quad (QzR, Qxz'Rp')$$

$$(3) \quad (QzpR, Qxz'pR)$$

$$(4) \quad (QzpR, Qxz'pRp')$$

$$(5) \quad (QzpR, Qxz'R)$$

$$(6) \quad (QzpR, Qxz'Rp')$$

$$(7) \quad (QzR, QxR)$$

$$(8) \quad (QzpR, QxR)$$

mit $z, z' \in V', x \in T, p, p' \in V \setminus (V' \cup T)$ und mit jeweils

beliebigen Wörtern $Q \in \Omega(T)$ und $R \in \Omega(V \setminus (V' \cup T))$,

außer in (1), (2) und (7), wo $R = \Lambda$ gesetzt werden darf.¹⁾

(c2) Für Π gilt außerdem:

$$(c21) \quad (QzR, Qxz'R) \in \Pi \Rightarrow (QzR, QxR) \in \Pi$$

$$(c22) \quad (QzpR, Qxz'R) \in \Pi \Rightarrow (QzpR, QxR) \in \Pi$$

Zur Einführung der von einem B-System erzeugten Sprache benötigen wir noch folgende Definitionen:

Def.: Seien q und r Wörter aus der freien Halbgruppe über V .

q erzeugt direkt r in $B=(V,V',T,\Pi,S)$, abgekürzt $q \rightarrow_B r$

(oder $q \rightarrow r$), wenn gilt:

$$(a) \quad q = QuRv \quad (b) \quad r = Qu'Rv' \quad (c) \quad (QuRv, Qu'Rv') \in \Pi$$

¹⁾ Q und R stellen keinen "Kontext" im üblichen Sinn dar (-abgesehen von der Möglichkeit, in (1), (2) und (7) $R = \Lambda$ zu setzen-), sondern ermöglichen lediglich eine "globale" Betrachtung der jeweiligen Ableitung.

Def.: q erzeugt r in B , abgekürzt $q \xrightarrow{*} r$ (oder $q \xrightarrow{*} r$), wenn gilt: Es gibt eine Folge von Ketten $r_0, r_1, \dots, r_s$ mit $s \geq 0, r_i \in \Omega(V)$ ($i=1, \dots, s$), so daß gilt:

$$(a) \ r_0 = q$$

$$(b) \ r_s = r$$

$$(c) \ (\forall i \ / \ 1 \leq i \leq s) (r_{i-1} \xrightarrow{*} r_i).$$

Die Folge $r_0, r_1, \dots, r_s$ heißt Ableitung von r aus q in B .

Def.: Die Menge $\{x \ / \ x \in \Omega(T) \wedge \exists \ x\}$ heißt die von B erzeugte Sprache und wird mit $L(B)$ bezeichnet.

Als Beispiel wollen wir ein B -System angeben, das die Menge aller Ketten aus $\Omega(\{a, b\}) \setminus \{\wedge\}$ erzeugt, in denen jeweils die gleiche Anzahl von a 's und b 's vorkommen. -Die so definierte Sprache ist im übrigen kontextfrei.-

$B = (\{a, b, u, v, z_0, z_1\}, \{z_0, z_1\}, \{a, b\}, \Pi, z_0)$, wobei

$$\begin{aligned} \Pi = \{ & (Qz_0R, Qaz_1Ru), (Qz_0R, Qbz_1Rv), (Qz_1R, Qaz_1Ru), \\ & (Qz_1R, Qbz_1Rv), (Qz_1uR, Qaz_1uRu), (Qz_1uR, Qbz_1R), \\ & (Qz_1uR, QbR), (Qz_1vR, Qbz_1vRv), (Qz_1vR, Qaz_1R), (Qz_1vR, QaR) \}. \end{aligned}$$

Daß die von B erzeugte Sprache obige Bedingung erfüllt, wollen wir nicht beweisen, man macht sich dies aber leicht an Beispielen klar.

Der Zusammenhang zwischen Pufferautomaten und B -Systemen wird in den beiden folgenden Sätzen aufgezeigt.

Satz 6-1 : Zu jedem einfachen Pufferautomaten $\mathcal{A}$ existiert ein B -System B , so daß gilt: $T(\mathcal{A}) = L(B)$.

Bew.: Zu jedem einfachen Pufferautomaten gibt es einen einfachen Pufferautomaten, der jeweils nur ein Symbol bei einem Zustandsübergang in den Puffer schreibt und der die gleiche Menge annimmt. Wir können uns also auf die

Betrachtung dieses Automatentyps beschränken. Außerdem können wir o.B.d.A. annehmen, daß Eingabemenge, Zustandsmenge und Pufferalphabet durchschnittsfremd sind.

Zunächst wird angegeben, wie zu einem solchen Pufferautomaten ein zugehöriges B-System konstruiert wird, und anschließend wird durch vollständige Induktion gezeigt, daß $T(\mathcal{B}) = L(B)$.

Gegeben sei $\mathcal{B} = (X, Z, P, B_2, \delta, z_0)$. Dann ist $B = (V, V', T, \Pi, S)$ folgendermaßen bestimmt:

$V := X \cup Z \cup P$; $V' := Z$; $T := X$; $S := z_0$ und Π wird entsprechend der Korrespondenz δ wie folgt konstruiert:

- 1) Falls $\delta[z, x, p] \supset \{(z', \wedge, b_{20})\}$ mit $p \in P$ in $\mathcal{B}$
 $\Rightarrow (Qz p R, Qx z' p R) \in \Pi$
- 2) Falls $\delta[z, x, p] \supset \{(z', p', b_{20})\}$ mit $p, p' \in P$ in $\mathcal{B}$
 $\Rightarrow (Qz p R, Qx z' p R p') \in \Pi$
- 3) Falls $\delta[z, x, \wedge] \supset \{(z', \wedge, b_{20})\}$
 oder $\delta[z, x, \wedge] \supset \{(z', \wedge, b_{21})\}$ in $\mathcal{B}$
 $\Rightarrow \{(Qz \wedge, Qx z' \wedge), (Qz \wedge, Qx \wedge)\} \subset \Pi$
- 4) Falls $\delta[z, x, \wedge] \supset \{(z', p', b_{20})\}$
 oder $\delta[z, x, \wedge] \supset \{(z', p', b_{21})\}$ mit $p' \in P$ in $\mathcal{B}$
 $\Rightarrow (Qz \wedge, Qx z' p') \in \Pi$
- 5) Falls $\delta[z, x, p] \supset \{(z', \wedge, b_{21})\}$ mit $p \in P$ in $\mathcal{B}$
 $\Rightarrow \{(Qz p R, Qx z' R), (Qz p R, Qx R)\} \subset \Pi$
- 6) Falls $\delta[z, x, p] \supset \{(z', p', b_{21})\}$ mit $p, p' \in P$ in $\mathcal{B}$
 $\Rightarrow (Qz p R, Qx z' R p') \in \Pi$

Π enthält nur die so konstruierten Produktionen.

Offensichtlich ist das so konstruierte System ein B-System, insbesondere erfüllt es die Bedingung der "gemeinsam auftretenden Produktionen" $[(c2\uparrow), (c22)]$.

Zum Beweis, daß $T(\mathcal{L}) = L(B)$, zeigen wir zunächst folgendes:

① Gilt in $\mathcal{L}$ $(z_0, \alpha\omega, \wedge) \vdash_{\mathcal{L}}^* (z, \omega, \pi)$ mit $\alpha \in \Omega(X) \setminus \{\wedge\}$, $\omega \in \Omega(X)$, $\pi \in \Omega(P)$, so gilt in B : $z_0 \xrightarrow{*} \alpha z \pi$.

Vollständige Induktion über die Länge von α :

$\alpha = x^1$; $|\alpha| = 1$: $(z_0, x^1\omega, \wedge) \vdash_{\mathcal{L}} (z, \omega, \pi)$

Um dies in $\mathcal{L}$ zu erreichen, muß es in $\mathcal{B}$ Relationen der Form

$$\delta[z_0, x^1, \wedge] \supset \{(z, \pi, b_{20})\}$$

bzw. $\delta[z_0, x^1, \wedge] \supset \{(z, \pi, b_{21})\}$ geben.

Nach 3) bzw. 4) (je nachdem, ob $\pi = \wedge$ oder $\pi \neq \wedge$) gibt es dann aber in B eine Produktion der Form $(z_0, x^1 z \pi)$, d.h., $z_0 \rightarrow x^1 z \pi$.

Der Beweis sei für Ketten der Länge $n-1$ schon erbracht:

$$\begin{aligned} (z_0, x^1 \dots x^{n-1} \omega, \wedge) &\vdash_{\mathcal{L}}^* (z', \omega, \pi') \\ &= z_0 \xrightarrow{*} x^1 \dots x^{n-1} z' \pi'. \end{aligned}$$

Gelte dann in $\mathcal{B}$ $(z_0, x^1 \dots x^{n-1} x^n \omega', \wedge) \vdash_{\mathcal{B}}^* (z; \omega', \pi)$, dann bedeutet dies

a) für $\pi' \neq \wedge$:

Es gibt in $\mathcal{B}$ Relationen der Form

$$\begin{aligned} \delta[z', x^n, p] &\supset \{(z, \wedge, b_{21})\}, \text{ falls } \pi' = p\pi \\ \text{bzw. } \delta[z', x^n, p] &\supset \{(z, p', b_{20})\}, \text{ falls } \pi \neq p\pi'' \wedge p \wedge \pi' = p\pi'' \\ \text{bzw. } \delta[z', x^n, p] &\supset \{(z, p', b_{21})\}, \text{ falls } \pi = \pi'' \wedge \pi' = p\pi'' \\ \text{bzw. } \delta[z', x^n, p] &\supset \{(z, \wedge, b_{20})\}, \text{ falls } \pi = \pi'' \wedge \pi' = p\pi'' \end{aligned}$$

Dann gibt es nach der Konstruktion von B folgende Produktionen:

$(Qz'pR, Qx^n zR)$ und $(Qz'pR, Qx^n R)$ nach 5)

bzw. $(Qz'pR, Qx^n zRp')$ nach 2)

bzw. $(Qz'pR, Qx^n zRp')$ nach 6)

bzw. $(Qz'pR, Qx^n zRp)$ nach 1)

$$b) \pi' = \Lambda \wedge \pi = \Lambda$$

$$\delta[z', x^n, \Lambda] \supset \{(z, \Lambda, b_{20})\}$$

$$\text{bzw. } \delta[z', x^n, \Lambda] \supset \{(z, \Lambda, b_{21})\}$$

$$\Rightarrow \text{in } B: (Qz' \wedge, Qx^n z \wedge) \text{ und } (Qz' \wedge, Qx^n \wedge) \text{ nach 3)}$$

$$c) \pi' = \Lambda \wedge \pi \neq \Lambda$$

$$\delta[z', x^n, \Lambda] \supset \{(z, p, b_{20})\}$$

$$\text{bzw. } \delta[z', x^n, \Lambda] \supset \{(z, p, b_{21})\}$$

$$\Rightarrow \text{in } B: (Qz' \wedge, Qx^n z p) \text{ nach 4)}$$

Nach Induktionsannahme galt:

$$z_0 \xrightarrow{*} x^1 \dots x^{n-1} z' \pi'.$$

Es gilt nun mit diesen Produktionen:

$$a) z_0 \xrightarrow{*} x^1 \dots x^{n-1} z' \pi' \rightarrow x^1 \dots x^n z \pi.$$

$$\text{und } z_0 \xrightarrow{*} x^1 \dots x^n \pi.$$

$$b) z_0 \xrightarrow{*} x^1 \dots x^n z \text{ und } z_0 \xrightarrow{*} x^1 \dots x^n.$$

$$c) z_0 \xrightarrow{*} x^1 \dots x^n z \pi.$$

Damit ist ① bewiesen.

Aus ① folgt aber unmittelbar, daß mit $w \in T(\mathcal{L})$ auch $w \in L(B)$ gilt, da nach b) eine Kette $w \in \Omega(T)$ in B erzeugbar ist.

② Andererseits folgt wegen der Konstruktion von Π auch:

$$z_0 \xrightarrow{*} \alpha z \pi \text{ mit } \alpha \in \Omega(X) \setminus \{\wedge\}, \pi \in \Omega(P) \text{ in } B$$

$$\Rightarrow (z_0, \alpha \omega, \wedge) \vdash_{\mathcal{L}}^* (z, \omega, \pi) \text{ in } \mathcal{B} \text{ für bel. } \omega \in \Omega(X).$$

Sei $w = x^1 \dots x^n \in L(B)$. Nach Definition existiert in B eine

Ableitung $z_0 \xrightarrow{*} w$. Dann muß es eine Ableitung der Form

$$a) z_0 \xrightarrow{*} x^1 \dots x^{n-1} z' p \rightarrow x^1 \dots x^n \text{ mit } z' \in Z, p \in P \text{ oder}$$

$$b) z_0 \xrightarrow{*} x^1 \dots x^{n-1} z' \rightarrow x^1 \dots x^n \text{ mit } z' \in Z$$

in B geben. In $\mathcal{B}$ gilt dann $(z_0, x^1 \dots x^{n-1} x^n, \wedge)$

$$\vdash_{\mathcal{L}}^* (z', x^n, p) \text{ bzw. } (z_0, x^1 \dots x^{n-1} x^n, \wedge) \vdash_{\mathcal{L}}^* (z', x^n, \wedge),$$

und es gibt in $\mathcal{B}$ Relationen der Form resp.

$$a) \quad \delta[z', x^n, p] \supset \{(z, \wedge, b_{21})\} \quad \text{nach 5)}$$

$$b) \quad \delta[z', x^n, \wedge] \supset \{(z, \wedge, b_{20})\} \quad \text{nach 3)}.$$

Daraus folgt $w \in T(\mathcal{L})$.

Satz 6-2: Zu jedem B-System B existiert ein einfacher Puffer-automat $\mathcal{L}$, so daß gilt: $L(B) = T(\mathcal{L})$.

Bew.: Sei ein B-System $B = (V, V', T, \Pi, S)$ gegeben.

Dann werde ein einfacher Pufferautomat $\mathcal{L} = (X, Z, P, B_2, \delta, z_0)$ folgendermaßen konstruiert: $X := T$; $Z := V' \cup \{\bar{z}\}$ mit $\bar{z} \notin V$; $P := V \setminus (V' \cup T)$; $B_2 := \{b_{20}, b_{21}\}$; $z_0 := S$ und δ wird entsprechend zu Π wie folgt konstruiert:

$$1) \ a) \ (QzR, QxR) \in \Pi \quad \text{mit } R \neq \wedge \Rightarrow \\ (\forall \pi \in P): \delta[z, x, \pi] \supset \{(z', \wedge, b_{20})\} \text{ in } \mathcal{L}$$

$$b) \ (Qz\wedge, Qxz'\wedge) \in \Pi \Rightarrow \\ \delta[z, x, \wedge] \supset \{(z', \wedge, b_{20})\} \text{ in } \mathcal{L}$$

$$2) \ a) \ (QzR, Qxz'Rp') \in \Pi \quad \text{mit } R \neq \wedge \Rightarrow \\ (\forall \pi \in P): \delta[z, x, \pi] \supset \{(z', p', b_{20})\} \text{ in } \mathcal{L}$$

$$b) \ (Qz\wedge, Qxz'p') \in \Pi \Rightarrow \\ \delta[z, x, \wedge] \supset \{(z', p', b_{20})\} \text{ in } \mathcal{L}$$

$$3) \ (QzpR, Qxz'pR) \in \Pi \Rightarrow \\ \delta[z, x, p] \supset \{(z', \wedge, b_{20})\} \text{ in } \mathcal{L}$$

$$4) \ (QzpR, Qxz'pRp') \in \Pi \Rightarrow \\ \delta[z, x, p] \supset \{(z', p', b_{20})\} \text{ in } \mathcal{L}$$

$$5) \ (QzpR, Qxz'R) \in \Pi \Rightarrow \\ \delta[z, x, p] \supset \{(z', \wedge, b_{21})\} \text{ in } \mathcal{L}$$

$$6) \ (QzpR, Qxz'Rp') \in \Pi \Rightarrow \\ \delta[z, x, p] \supset \{(z', p', b_{21})\} \text{ in } \mathcal{L}$$

$$7) \ a) \ (QzR, QxR) \in \Pi \quad \text{mit } R \neq \wedge \Rightarrow \\ (\forall \pi \in P): \begin{cases} \delta[z, x, \pi] \supset \{(z', \wedge, b_{20})\} & \text{falls } (QzR, QxR) \in \Pi \\ \delta[z, x, \pi] \supset \{(\bar{z}, \wedge, b_{20})\} & \text{sonst} \end{cases}$$

$$b) (Qz\wedge, Qx\wedge) \in \Pi \Rightarrow$$

$$\begin{cases} \delta[z, x, \wedge] \supset \{(z', \wedge, b_{20})\} & \text{falls } (Qz\wedge, Qx\wedge) \in \Pi \\ \delta[z, x, \wedge] \supset \{(\bar{z}, \wedge, b_{20})\} & \text{sonst} \end{cases}$$

$$8) (QzpR, QxR) \in \Pi \Rightarrow$$

$$\begin{cases} \delta[z, x, p] \supset \{(z', \wedge, b_{21})\} & \text{falls } (QzpR, Qx\wedge R) \in \Pi \\ \delta[z, x, p] \supset \{(\bar{z}, \wedge, b_{21})\} & \text{sonst} \end{cases}$$

In $\mathcal{S}$ sind auch nur die so eingeführten Relationen enthalten.

Analog wie im Beweis von Satz 6-1 zeigt man durch vollständige Induktion über die Länge der Kette α leicht, daß aus $S \xrightarrow{*} \alpha z \pi$ in B folgt: $(z_0, \alpha \omega, \wedge) \vdash_{\mathcal{S}}^* (z, \omega, \pi)$ für $\alpha \in \Omega(T)$, $\pi \in \Omega(V \setminus (V' \cup T))$ und für beliebige $\omega \in \Omega(X)$. Man sieht unmittelbar, daß aus $w \in L(B)$ folgt:

$$(z_0, w, \wedge) \vdash_{\mathcal{S}}^* (z, \wedge, \wedge) \text{ bzw. } (z_0, w, \wedge) \vdash_{\mathcal{S}}^* (\bar{z}, \wedge, \wedge),$$

d.h. $w \in T(\mathcal{S})$.

Aufgrund der Konstruktion von δ gilt aber auch wieder:

$$\text{Aus } (z_0, \alpha \omega, \wedge) \vdash_{\mathcal{S}}^* (z, \omega, \pi) \text{ folgt } S \xrightarrow{*} \alpha z \pi.$$

$$\text{Ist } x^1 \dots x^n = w \in T(\mathcal{S}), \text{ d.h. } (z_0, w, \wedge) \vdash_{\mathcal{S}}^* (z, \wedge, \wedge),$$

dann kann dies nur entstanden sein durch direkten Übergang $(z', x^n, \wedge) \vdash_{\mathcal{S}} (z, \wedge, \wedge)$ oder durch (z', x^n, p)

$\vdash_{\mathcal{S}} (z, \wedge, \wedge)$, da nur solche Relationen im konstruierten Pufferautomaten $\mathcal{S}$ vorkommen, die höchstens ein Symbol im Puffer bei einem Zustandsübergang löschen. Diese Übergänge werden resp. durch Relationen $\delta[z', x^n, \wedge] \supset \{(z, \wedge, b_{20})\}$ und $\delta[z', x^n, p] \supset \{(z, \wedge, b_{21})\}$ bewirkt. Wegen der Bedingung ((c21), (c22)) des gleichzeitigen Auftretens von Produktionen des Typs 1) und 7) bzw. 5) und 8) in einem B-System gibt es deshalb auch Produktionen der Form $(Qz'\wedge, Qx^n\wedge)$ und $(Qz'pR, QxR)$, und daraus folgt $w \in L(B)$.

Um einen Zusammenhang mit den einfachen deterministischen Pufferautomaten herstellen zu können, führen wir etwas eingeschränkte B-Systeme ein.

Def.: Ein B-System (V, V', T, Π, S) heißt teildeterminiert, wenn folgendes gilt:

- (1) Für alle Paare von Produktionen $[(Q_1 u_1 R_1 v_1, Q_1 u'_1 R_1 v'_1), (Q_2 u_2 R_2 v_2, Q_2 u'_2 R_2 v'_2)] \in \Pi \times \Pi$ des Typs (1)-(6) bzw. des Typs (7), (8) muß (getrennt) gelten:

$$(u_1, \bar{u}'_1) \neq (u_2, \bar{u}'_2).$$

Dabei sind

- a) für Produktionen vom Typ (1)-(6)

$$u'_i = \begin{cases} \bar{u}'_i z' p & \text{falls } p \neq \Lambda \\ \bar{u}'_i z' & \text{falls } p = \Lambda \end{cases} \quad (i=1,2)$$

- b) für Produktionen vom Typ (7), (8)

$$u'_i = \begin{cases} \bar{u}'_i p & \text{falls } p \neq \Lambda \\ \bar{u}'_i & \text{falls } p = \Lambda \end{cases} \quad (i=1,2)$$

- (2) Das B-System darf keine Produktionenpaare der Form

$(QzR, QxRv')$ und $(Q'zpR', Q'xp''R'v'')$ bzw. der Form

$(QzR, Qxz'Rv')$ und $(Q'zpR', Q'xz''p'R'v'')$ mit $R \neq \Lambda$,

$z', z'' \in V', p \in V \setminus (V' \cup T), p'' \in V \setminus (V' \cup T) \cup \{\Lambda\}$ enthalten.

Es soll bemerkt werden, daß die Bedingung der Teildeterminiertheit trotz ähnlicher Bedeutung nicht mit der Bedingung der Monogenität von kombinatorischen Systemen zusammenfällt.

Damit können wir nun die deterministischen Puffersprachen auch folgendermaßen charakterisieren:

Korollar 6-1: Zu jeder deterministischen Puffersprache $\mathcal{L}$ gibt es ein teildeterminiertes B-System B, so daß gilt:

$$\mathcal{L} = L(B).$$

Der Beweis folgt unmittelbar aus der Tatsache, daß es nach De-

finition der deterministischen Puffersprachen zu jeder von ihnen einen einfachen deterministischen Pufferautomaten $\mathcal{A}$, gibt, der sie annimmt, daß der einfache deterministische Pufferautomat durch die Abbildung von $Z \times X \times \bar{P}$ in $Z \times \bar{P} \times B_2$ bestimmt wird und aus der Konstruktion von Satz 6-1.

Die Umkehrung gilt ebenfalls:

Korollar 6-2: Zu jedem teildeterminierten B-System B gibt es eine deterministische Puffersprache $\mathcal{L}$, so daß gilt: $L(B) = \mathcal{L}$.

Der Beweis ergibt sich nach der Definition der deterministischen Puffersprachen daraus, daß sich nach Satz 6-2 zu einem teildeterminierten B-System ein Pufferautomat konstruieren läßt, der i.a. allerdings noch keinen deterministischen Pufferautomaten darstellt, da er nicht durch eine Abbildung, sondern nur durch eine eindeutige Korrespondenz bestimmt ist. Diese läßt sich dann aber trivialerweise (evtl. unter Erweiterung der Zustandsmenge) zu einer Abbildung ergänzen.

7. Entscheidbarkeitsfragen bei Puffersprachen

Zu den charakterisierenden Eigenschaften von Sprachen gehören neben ihrer Einordnung in die CHOMSKY -Hierarchie und ihr Verhalten gegenüber gewissen Operationen auch die Möglichkeit bzw. Unmöglichkeit bestimmte Fragen, wie z.B. die nach der Gleichheit zweier beliebiger Sprachen dieser Klasse, algorithmisch entscheiden zu können.

Wir werden in diesem Abschnitt u.a. zeigen, daß es rekursiv unentscheidbar ist, ob der Durchschnitt zweier deterministischer Puffersprachen leer ist. Den Beweis für die Unentscheidbarkeit der Gleichheit von Puffersprachen werden wir lediglich für nichtdeterministische Puffersprachen erbringen.

Alle Unentscheidbarkeitsbeweise für Puffersprachen beruhen auf der Unentscheidbarkeit des berühmten POST'schen Korrespondenzproblems, das wir zitieren werden. Sie werden in Analogie zu den entsprechenden Sätzen für kontextfreie Sprachen (s.z.B. GINSBURG(1966) und LANDWEBER (1964)) verlaufen, die wiederum im wesentlichen auf einen Beweisgedanken von RABIN-SCOTT (1959) zurückzuführen sind.

POST'scher Korrespondenzsatz (POST (1946)) :

Sei X eine mindestens zweielementige, endliche Menge, und seien für eine beliebige natürliche Zahl n $(x_1, \dots, x_n)$ und $(y_1, \dots, y_n)$ beliebige n -tupel von nicht-leeren Wörtern aus $\Omega(X)$. Dann ist es unentscheidbar, ob eine nichtleere Folge von Indizes $i_1, \dots, i_k$ ($1 \leq i_v \leq n$, $v = 1, \dots, k$) existiert, so daß gilt:

$$x_{i_1} \dots x_{i_k} = y_{i_1} \dots y_{i_k}.$$

Bevor wir zu den Unentscheidbarkeitsresultaten gelangen, wollen wir zunächst ein entscheidbares Problem formulieren:

Satz 7-1: Es ist entscheidbar, ob ein beliebiges Wort w Element einer beliebigen (deterministischen oder nicht-deterministischen) Puffersprache L ist.

Bew.: Ist $w = \Lambda$, dann ist w nach Definition der Puffersprachen nicht Element irgendeiner Puffersprache, also auch nicht von L .

Nach Definition der Puffersprachen gibt es zu L einen einfachen Pufferautomaten $\mathcal{B} = (X, Z, P, B_2, \delta, z_0)$, der gerade L annimmt. δ ist dabei eine Korrespondenz von $Z \times X \times \bar{P}$ in $Z \times \bar{P} \times B_2$. Sei $l = |w| > 0$. Dann gibt es, da immer von der Konfiguration (z_0, w, Λ) ausgegangen wird und bei jedem Übergang genau ein Symbol aus w abgearbeitet wird, in $\mathcal{B}$ höchstens $(|Z| \cdot (|P| + 1) \cdot |B_2|)^l$ Übergangsfolgen bis zur Abarbeitung von w . Daraus folgt die Behauptung.

Zum Beweis der Unentscheidbarkeitssätze benötigen wir einige deterministische Puffersprachen, die wir nun einführen wollen. - Wir betrachten in diesem Abschnitt jeweils Sprachen über einem Alphabet mit drei oder mehr Elementen. -

Lemma 7-1: Für jedes n -tupel ($n \geq 1$) $x = (x_1, \dots, x_n)$ von nicht-

leeren Wörtern aus $\Omega(\{a, b\})$ ist die Menge

$$L(x) := \{ba^k ba^{i_1} \dots ba^{i_k} cx_{i_k} \dots x_{i_1} / k \geq 1, 1 \leq i_\mu \leq n \text{ für } \mu = 1, \dots, k\}$$

eine deterministische Puffersprache.

Bew.: Sei $x = (x_1, \dots, x_n)$ gegeben mit $x_1, \dots, x_n \in \Omega(\{a, b\})$

($v = 1, \dots, n$). Wir wollen dann ein teildeterminiertes

B-System $B_0 = (V_0, V'_0, T_0, \Pi_0, S_0)$ angeben, für das gilt:

$L(B_0) = L(x)$. Nach Korollar 6-2 folgt daraus, daß $L(x)$

eine deterministische Puffersprache ist.

$$\text{Sei } V_0 := V'_0 \cup T_0 \cup \{\pi_0, \pi_1, \dots, \pi_n\}$$

$$V'_0 := \{z, z_0, \dots, z_{n+1}, z_1^1, \dots, z_1^{l_1-1}, \dots, z_n^1, \dots, z_n^{l_n-1}\}$$

$$T_0 := \{a, b, c\} \quad S_0 := z$$

$$\pi_0 := \pi_1 \cup \pi_2 \cup \pi_3$$

$$\text{Dabei sind: } \pi_1 := \{(Qz_R, Qbz_0 R \pi_0)\}$$

$$\pi_2 := \{(Qz_0 R, Qaz_1 R), (Qz_0 R, QaR),$$

$$(Qz_1 R, Qaz_2 R), (Qz_1 R, QaR), (Qz_1 R, Qbz_0 R \pi_1),$$

$$(Qz_2 R, Qaz_3 R), (Qz_2 R, QaR), (Qz_2 R, Qbz_0 R \pi_2),$$

.

$$(Qz_{n-1} R, Qaz_n R), (Qz_{n-1} R, QaR), (Qz_{n-1} R, Qbz_0 R \pi_{n-1}),$$

$$(Qz_n R, Qbz_0 R \pi_n),$$

$$(Qz_{n+1} \pi_1 R, Qx_1^1 z_{n+1} R), (Qz_{n+1} \pi_1 R, Qx_1^1 R) \quad \text{falls } l_1 = 1$$

$$(Qz_{n+1} \pi_1 R, Qx_1^1 z_1^1 \pi_1 R),$$

$$(Qz_1^1 \pi_1 R, Qx_1^1 z_1^1 \pi_1 R),$$

.

$$(Qz_1^{l_1-1} \pi_1 R, Qx_1^{l_1-1} z_{n+1} R), (Qz_1^{l_1-1} \pi_1 R, Qx_1^{l_1-1} R)$$

$$(Qz_{n+1} \pi_2 R, Qx_2^1 z_{n+1} R), (Qz_{n+1} \pi_2 R, Qx_2^1 R) \quad \text{falls } l_2 = 1$$

$$(Qz_{n+1} \pi_2 R, Qx_2^1 z_2^1 \pi_2 R),$$

.

$$(Qz_2^{l_2-1} \pi_2 R, Qx_2^{l_2-1} z_{n+1} R), (Qz_2^{l_2-1} \pi_2 R, Qx_2^{l_2-1} R)$$

.

$$(Qz_{n+1} \pi_n R, Qx_n^1 z_{n+1} R), (Qz_{n+1} \pi_n R, Qx_n^1 R) \quad \text{falls } l_n = 1$$

$$(Qz_{n+1} \pi_n R, Qx_n^1 z_n^1 \pi_n R),$$

.

$$(Qz_n^{l_n-1} \pi_n R, Qx_n^{l_n-1} z_{n+1} R), (Qz_n^{l_n-1} \pi_n R, Qx_n^{l_n-1} R) \quad \}$$

$$\pi_3 := \{(Qz_1 \pi_0 R, Qcz_{n+1} R \pi_1), (Qz_2 \pi_0 R, Qcz_{n+1} R \pi_2), \dots$$

$$(Qz_n \pi_0 R, Qcz_{n+1} R \pi_n)\}$$

Man überzeugt sich leicht, daß B_0 die Bedingungen für ein teildeterminiertes B-System erfüllt und daß $L(B_0) = L(x)$.

Lemma 7-2: Für jedes Paar von n-tupeln ($n \geq 1$) $x=(x_1, \dots, x_n)$
und $y=(y_1, \dots, y_n)$ von nichtleeren Wörtern aus
 $\Omega(\{a, b\})$ ist die Menge

$$L_1 := \{ba^{i_1} \dots ba^{i_k} cx_{i_1} \dots x_{i_n} dba^{j_1} \dots ba^{j_m} cy_{j_1} \dots y_{j_m} /$$

$$/ k, m \geq 1, 1 \leq i_\mu, j_\mu \leq n \text{ für } \mu=1, \dots, k \text{ und}$$

$$x=1, \dots, m\}$$

eine deterministische Puffersprache.

Bew.: Seien $x=(x_1, \dots, x_n)$ mit $x_v = x_v^1 \dots x_v^{l_v} \in \Omega(\{a, b\}) \setminus \{\Lambda\}$

($v=1, \dots, n$) und $y=(y_1, \dots, y_n)$ mit $y_v = y_v^1 \dots y_v^{p_v} \in \Omega(\{a, b\}) \setminus \{\Lambda\}$
($v=1, \dots, n$) gegeben.

Wir werden wieder ein teildeterminiertes B-System

$B=(V, V', T, \Pi, S)$ angeben, so daß gilt: $L(B)=L_1=L(x)dL(y)$.

Die Produktionenmenge, um die Menge $L(x)$ zu erzeugen,

wird im wesentlichen die gleiche sein wie in Lemma 7-1.

Modifikationen sind jedoch nötig, um sicherzustellen,

daß $L(y)$ nicht leer sein darf. Um $L(y)$ zu erzeugen, werden

die Produktionsmengen Π_1, Π_2 und Π_3 aus Lemma 7-1 bis
auf eine Änderung der Zustandsbezeichnungen benutzt.

$$V := V \cup T \cup \{\pi_0, \pi_1, \dots, \pi_n, \bar{\pi}_1, \dots, \bar{\pi}_n\}$$

$$V' := V'_0 \cup \{\bar{z}, \bar{z}_0, \dots, \bar{z}_{n+1}, \bar{z}_1^1, \dots, \bar{z}_1^{p_1-1}, \dots, \bar{z}_n^1, \dots, \bar{z}_n^{p_n-1}\}$$

$$T := \{a, b, c, d\}; S := z$$

$$\Pi := \Pi_1 \cup \Pi_2 \cup \bar{\Pi}_2 \cup \bar{\Pi}_3 \cup \Pi_4$$

Dabei sind:

$$\Pi_4 := \left\{ \begin{array}{l} (Qz_1 \pi_0 R, Qcz_{n+1} R \bar{\pi}_1), \dots, (Qz_n \pi_0 R, Qcz_{n+1} R \bar{\pi}_n), \\ (Qz_{n+1} \bar{\pi}_1 R, Qx_1^1 z_{n+1} R \pi_0) \quad \text{falls } l_1=1 \\ (Qz_{n+1} \bar{\pi}_1 R, Qx_1^1 z_1^1 \bar{\pi}_1 R) \\ \dots \dots \dots \\ (Qz_1^{l_1-1} \bar{\pi}_1 R, Qx_1^{l_1} z_{n+1} R \pi_0) \end{array} \right\} \quad \text{falls } l_1 > 1$$

$$\begin{aligned}
 & (Qz_{n+1} \bar{\pi}_n R, Qx_n^1 z_{n+1} R \pi_0) \quad \text{falls } l_n = 1 \\
 & \left. \begin{aligned}
 & (Qz_{n+1} \bar{\pi}_n R, Qx_n^1 z_n^1 \bar{\pi}_n R) \\
 & \dots \dots \dots \\
 & (Qz_n^{l_n-1} \bar{\pi}_n R, Qx_n^{l_n} z_{n+1} R \pi_0)
 \end{aligned} \right\} \quad \text{falls } l_n > 1 \\
 & (Qz_{n+1} \pi_0 R, Qd\bar{z} \pi_0 R), (Q\bar{z} R, Qb\bar{z}_0 R), (Q\bar{z} R, QbR) \} \\
 & \bar{\pi}_2 := \{ (Q\bar{z}_0 R, Qa\bar{z}_1 R), (Q\bar{z}_0 R, QaR), \\
 & (Q\bar{z}_1 R, Qa\bar{z}_2 R), (Q\bar{z}_1 R, QaR), (Q\bar{z}_1 R, Qb\bar{z}_0 R \pi_1), \\
 & \dots \dots \dots \\
 & (Q\bar{z}_n R, Qb\bar{z}_0 R \pi_n), \\
 & (Q\bar{z}_{n+1} \pi_1 R, Qy_1^1 \bar{z}_{n+1} R), (Q\bar{z}_{n+1} \pi_1 R, Qy_1^1 R) \quad \text{falls } p_1 = 1 \\
 & \left. \begin{aligned}
 & (Q\bar{z}_{n+1} \pi_1 R, Qy_1^1 \bar{z}_1^1 \pi_1 R), \\
 & \dots \dots \dots \\
 & (Q\bar{z}_1^{p_1-1} \pi_1 R, Qy_1^{p_1} \bar{z}_{n+1} R), (Q\bar{z}_1^{p_1-1} \pi_1 R, Qy_1^{p_1} R)
 \end{aligned} \right\} \quad \text{falls } p_1 > 1 \\
 & \dots \dots \dots \\
 & (Q\bar{z}_{n+1} \pi_n R, Qy_n^1 \bar{z}_{n+1} R), (Q\bar{z}_{n+1} \pi_n R, Qy_n^1 R) \quad \text{falls } p_n = 1 \\
 & \left. \begin{aligned}
 & (Q\bar{z}_{n+1} \pi_n R, Qy_n^1 \bar{z}_n^1 R), \\
 & \dots \dots \dots \\
 & (Q\bar{z}_n^{p_n-1} \pi_n R, Qy_n^{p_n} \bar{z}_{n+1} R), (Q\bar{z}_n^{p_n-1} \pi_n R, Qy_n^{p_n} R)
 \end{aligned} \right\} \quad \text{falls } p_n > 1 \\
 & \bar{\pi}_3 := \{ (Q\bar{z}_1 \pi_0 R, Qc\bar{z}_{n+1} R \pi_1), \dots, (Q\bar{z}_n \pi_0 R, Qc\bar{z}_{n+1} R \pi_n) \}
 \end{aligned}$$

Man überzeugt sich wieder leicht, daß B ein teildeterminiertes B-System ist. Daß die oben angegebene Sprache erzeugt wird, wird deutlich, wenn man beachtet, daß $\bar{\pi}_1$ im wesentlichen dazu eingeführt wurde, damit $L(y)$ nicht leer werden kann und daß sich $\bar{\pi}_2$ und $\bar{\pi}_3$ von $\bar{\pi}_1$ resp. $\bar{\pi}_3$ lediglich durch die geänderten Zustandsbezeichnungen, das Ersetzen von x_v^s durch y_v^s und von l_v durch p_v unterscheiden, da natürlich für $v = 1, \dots, n$ die Längen der jeweiligen Ketten x_v und y_v nicht übereinzustimmen brauchen.

Lemma 7-3: Die Menge $L_2 := \{wdw / w \in \Omega(\{a,b,c\}) \setminus \{\Lambda\}\}$ ist eine deterministische Puffersprache.

Bew.: Ein teildeterminiertes B-System $B=(V,V',T,\Pi,S)$, das L_2 erzeugt, ist folgendermaßen gegeben:

$$V := V' \cup T \cup \{u,v,w\} ; \quad V' := \{z_0, z_1, z_2\} ; \quad T := \{a,b,c,d\} ;$$

$$S := z_0$$

$$\begin{aligned} \Pi := & \{ (Qz_0R, Qaz_1Ru), (Qz_0R, Qbz_1Rv), (Qz_0R, Qcz_1Rw), \\ & (Qz_1R, Qaz_1R), (Qz_1R, Qbz_1Rv), (Qz_1R, Qcz_1Rw), (Qz_1R, Qdz_2R), \\ & (Qz_1R, QdR), (Qz_2uR, Qaz_2R), (Qz_2uR, QaR), (Qz_2vR, Qbz_2R), \\ & (Qz_2vR, QbR), (Qz_2wR, Qcz_2R), (Qz_2wR, QcR) \} \end{aligned}$$

Man sieht unmittelbar, daß gilt: $L(B) = L_2$.

Satz 7-2: Für deterministische Puffersprachen L_1 und L_2 sind die folgenden Probleme rekursiv unentscheidbar:

- (a) Ist $L_1 \cap L_2 = \emptyset$?
- (b) Ist $L_1 \cap L_2$ eine endliche Menge ?
- (c) Ist $L_1 \cap L_2$ eine reguläre Sprache ?
- (d) Ist $L_1 \cap L_2$ eine kontextfreie Sprache ?

Bew.: Um die Beweise zu führen, betrachten wir den Durchschnitt der oben eingeführten deterministischen Puffersprachen L_1 und L_2 : Offensichtlich gilt:

$$\begin{aligned} L_1 \cap L_2 &= \{ba^k \dots ba^k cx_i \dots x_k dba^k \dots ba^k y_j \dots y_m / \\ &\quad / k, m \geq 1, 1 \leq i \leq n, 1 \leq j \leq n \text{ für } \mu=1, \dots, k, x=1, \dots, m\} \\ &\cap \{wdw / w \in \Omega(\{a,b,c\}) \setminus \{\Lambda\}\} \\ &= \{ba^k \dots ba^k cx_i \dots x_k dba^k \dots ba^k cy_i \dots y_k / k \geq 1 \\ &\quad \wedge 1 \leq i \leq n \text{ für } \mu=1, \dots, k, x_i \dots x_k = y_i \dots y_k\}. \end{aligned}$$

Die Beweise verlaufen derart, daß gezeigt wird, daß die Entscheidbarkeit der Probleme (a)-(d) jeweils die Entscheidbarkeit des Korrespondenzproblems für das Paar von n-tupeln $(x_1, \dots, x_n)$ und $(y_1, \dots, y_n)$ implizieren

würde. Da aber für jedes Paar von n -tupeln (für beliebige $n \geq 1$) die entsprechenden Mengen L_1 und L_2 deterministische Puffersprachen sind, ergäbe sich damit ein Widerspruch zum POST'schen Korrespondenzsatz. Wir wollen nun auf die einzelnen Probleme ausführlich eingehen:

- (a) Wäre es entscheidbar, ob $L_1 \cap L_2 = \emptyset$, so wäre auch das Korrespondenzproblem entscheidbar. Gälte nämlich $L_1 \cap L_2 \neq \emptyset$, d.h., enthielte der Durchschnitt von L_1 und L_2 mindestens ein Element, so bedeutete dies, daß eine Indexfolge $i_1, \dots, i_k$ existierte, für die $x_{i_1} \dots x_{i_k} = y_{i_1} \dots y_{i_k}$, während es im Fall $L_1 \cap L_2 = \emptyset$ keine solche Indexfolge gäbe.
- (b) Ist der Durchschnitt von L_1 und L_2 eine endliche Menge, so folgt daraus, daß er leer ist. Gibt es nämlich eine Indexfolge $i_1 \dots i_k$, für die $x_{i_1} \dots x_{i_k} = y_{i_1} \dots y_{i_k}$, dann gibt es unendlich viele Indexfolgen $j_1 \dots j_1$, für die $x_{j_1} \dots x_{j_1} = y_{j_1} \dots y_{j_1}$ gilt und zwar $(i_1 \dots i_k)^n$ für beliebiges n . D.f., daß im Falle, daß $L_1 \cap L_2$ eine endliche Menge wäre, das Korrespondenzproblem keine Lösung hätte, während es im andern Fall eine, bzw. sogar unendlich viele Lösungen hätte.
- (c), (d) Enthält der Durchschnitt $L_1 \cap L_2$ ein Element, so enthält er, wie wir oben gesehen haben, unendlich viele Elemente und ist, da $\{w \mid w \in \Omega(\{a, b, c\}) \setminus \{\Lambda\}\}$ bekanntlich keine kontextfreie Sprache ist, ebenfalls keine kontextfreie Sprache und deshalb a fortiori keine reguläre Sprache.
- $L_1 \cap L_2$ ist nur dann eine reguläre und auch kontextfreie Sprache, wenn $L_1 \cap L_2 = \emptyset$.

Die Möglichkeit zu entscheiden, ob $L_1 \cap L_2$ reguläre bzw. kontextfreie Sprache ist, würde damit auch die Entscheidbarkeit des Korrespondenzproblems implizieren.

Für nichtdeterministische Puffersprachen können wir die Unentscheidbarkeit der Enthaltenheitsbeziehung nachweisen. Wir benötigen dazu noch einige Lemmata.

Wie wir in Abschnitt 5 sahen, stellt die Vereinigung zweier deterministischer Puffersprachen nicht notwendigerweise wieder eine deterministische Puffersprache dar. Es gilt aber:

Lemma 7-4: Die Klasse der nichtdeterministischen Puffersprachen ist bezüglich der Vereinigung abgeschlossen.

Bew.: Seien L_1 und L_2 beliebige nichtdeterministische Puffersprachen. Dann gibt es nach Satz 6-1 B-Systeme $B_i = (V_i, V'_i, T_i, \Pi_i, S_i)$ ($i=1,2$), so daß gilt: $L(B_1) = L_1$ und $L(B_2) = L_2$. O.B.d.A. können wir voraussetzen, daß $(V_1 \setminus T_1) \cap (V_2 \setminus T_2) = \emptyset$. Wir konstruieren dann ein B-System $B = (V, V', T, \Pi, S)$ auf folgende Weise:

$$\begin{aligned} V &:= V_1 \cup V_2 \cup \{S\} \text{ mit } S \notin V_1 \cup V_2; V' := V'_1 \cup V'_2 \cup \{S\}; \\ T &:= T_1 \cup T_2; \Pi := \Pi_1 \cup \Pi_2 \cup \{(S, xzp) / (QS_1R, QxzRp) \in \Pi_1 \\ &\quad \wedge R \in \Omega(V_1 \setminus (V'_1 \cup T_1)) \wedge p \in V_1 \setminus (V'_1 \cup T_1) \cup \{\lambda\} \\ &\quad \cup \{(S, x'z'p') / (QS_2R, Qx'z'Rp') \in \Pi_2 \wedge R \in \Omega(V_2 \setminus (V'_2 \cup T_2)) \\ &\quad \wedge p' \in V_2 \setminus (V'_2 \cup T_2) \cup \{\lambda\}\} \end{aligned}$$

Man sieht unmittelbar, daß gilt $L(B) = L_1 \cup L_2$ (-insbesondere auch, daß $L(B)$ leer ist, falls dies L_1 und L_2 sind.-)

Im folgenden müssen wir noch nachweisen, daß die Komplemente der Sprachen $L(x)$ und $L(y)$ nichtdeterministische Puffersprachen sind.

Lemma 7-5: Für jedes n-tupel ($n \geq 1$) $x = (x_1^{i_1} \dots x_n^{i_n}, \dots, x_n^{j_1} \dots x_n^{j_n})$ von nichtleeren Wörtern aus $\Omega(\{a, b\})$ ist die Menge $\bar{L}(x) = (\Omega(\{a, b, c\}) \setminus \{\Lambda\}) \setminus L(x)$ eine nichtdeterministische Puffersprache.

Bew.: Wir wollen das sehr umfangreiche B-System, das die Sprache $\bar{L}(x)$ erzeugt, nicht im einzelnen angeben, sondern in weiten Teilen nur mit Worten skizzieren.

Einige Komponenten der Sprache werden wir ähnlich wie in einem entsprechenden Beweis bei GINSBURG (1966) als reguläre Sprachen schreiben (in der dort gebräuchlichen, sich aber auch wohl selbst erklärenden Form -abgesehen von der Bezeichnung der freien Halbgruppe mit " Ω " statt " $*$ "-):

Alle Ketten aus $\Omega(\{a, b, c\}) \setminus \{\Lambda\}$, die kein c oder mindestens zwei c enthalten, werden dargestellt durch

$$R_1 := \Omega(\{a, b\}) \setminus \{\Lambda\} \cup [\Omega(\{a, b\})\{c\}]^2 \Omega(\{a, b\})$$

Alle Ketten aus $\Omega(\{a, b, c\}) \setminus \{\Lambda\}$, die mit a oder c beginnen oder mit c enden oder mindestens c^2 enthalten, werden dargestellt durch

$$R_2 := \{a, c\} \Omega(\{a, b, c\}) \cup \Omega(\{a, b, c\})\{c\} \cup \Omega(\{a, b, c\}) \cdot \{c^2\} \Omega(\{a, b, c\})$$

Alle Ketten aus $\Omega(\{a, b, c\}) \setminus \{\Lambda\}$, die entweder mindestens b^2 vor c oder b unmittelbar links von c enthalten, werden dargestellt durch :

$$R_3 := \Omega(\{a, b\})\{b^2\} \Omega(\{a, b\})\{c\} \Omega(\{a, b, c\}) \cup \Omega(\{a, b\})\{bc\} \Omega(\{a, b, c\})$$

Alle Ketten aus $\Omega(\{a, b, c\}) \setminus \{\Lambda\}$, die irgendein a^h mit $h > n$ vor c enthalten, werden dargestellt durch:

$$R_4 := \Omega(\{a, b\})\{a^{n+1}\} \Omega(\{a, b\})\{c\} \Omega(\{a, b, c\})$$

Alle Ketten aus $\Omega(\{a,b,c\}) \setminus \{A\}$ der Form

$\{ba^{i_1} \dots ba^{i_k} cw \mid k \geq 1, 1 \leq i_\mu \leq n \text{ für } \mu = 1, \dots, k$

$\wedge w \in \Omega(\{a,b\}) \wedge w \neq x_i, \dots, x_n, \wedge |w| \leq |x_i|, \dots, |x_n|\}$

werden durch das im folgenden skizzierte B-System

$\bar{B} = (\bar{V}, \bar{V}', \bar{T}, \bar{\Pi}, \bar{S})$ erzeugt:

$\bar{V} \setminus (\bar{V}' \cup \bar{T}) := \{\pi_0, \pi_1, \dots, \pi_n, \bar{\pi}_1, \dots, \bar{\pi}_n\}$

$\bar{T} := \{a, b\}$; $\bar{S} := z$

$\bar{\Pi} \supset \{(Qz_R, Qbz_0 R \pi_0), (Qz_0 R, Qaz_1 R), (Qz_0 R, QaR),$
 $(Qz_1 R, Qaz_2 R), (Qz_1 R, QaR), (Qz_1 R, Qbz_0 R \pi_1), (Qz_1 R, Qbz_0 R),$
 $(Qz_1 R, QbR), (Qz_1 \pi_0 R, Qcz_{n+1} R \bar{\pi}_1),$
 $\dots$
 $(Qz_{n-1} R, Qaz_n R), (Qz_{n-1} R, QaR), (Qz_{n-1} R, Qbz_0 R \pi_{n-1}),$
 $(Qz_{n-1} R, Qbz_0 R), (Qz_{n-1} R, QbR), (Qz_{n-1} \pi_0 R, Qcz_{n+1} R \bar{\pi}_{n-1}),$
 $(Qz_n R, Qbz_0 R \pi_n), (Qz_n R, Qbz_0 R), (Qz_n R, QbR), (Qz_n \pi_0 R, Qcz_{n+1} R \bar{\pi}_n)\}$

Mit dieser Teilmenge der Produktionen von $\bar{B}$ werden

$\{ba^{i_1} \dots ba^{i_k} c \mid k \geq 1, 1 \leq i_\mu \leq n \text{ für } \mu = 1, \dots, k\}$ und
 $\{ba^{i_1} \dots ba^{i_s} cz_{n+1} \bar{\pi}_{i_s} \dots \pi_{i_{s+1}} \bar{\pi}_{i_{s+1}} \mid s \geq 0, j_r < j_{r+1} \text{ für } r=1, \dots, s\}$

erzeugt. Von den Elementen dieser letztgenannten Menge ausgehend, verläuft nun der weitere Erzeugungsprozeß folgendermaßen:

Bei Vorliegen von $Qz_{n+1} \pi_i R$ ($1 \leq i \leq n$) werden entweder

- alle verschiedenen Ketten $x_i^j \in \Omega(\{a,b\})$ mit $|x_i^j| \leq l_1$ erzeugt: Es wird nach $Q x_i^j \bar{z} R$ bzw. $Q x_i^j R$ übergegangen oder
- es werden alle verschiedenen Ketten $\bar{x}_i^j \in \Omega(\{a,b\})$ mit $|\bar{x}_i^j| = l_1$ und $\bar{x}_i^j \neq x_i$ erzeugt: Es wird nach $Q \bar{x}_i^j \bar{z}_{n+1} R$ bzw. $Q \bar{x}_i^j R$ übergegangen.

Die Symbole $\bar{\pi}_i$ ($1 \leq i \leq n$), die gleichzeitig mit c erzeugt werden, dienen dazu zu verhindern, daß eine Kette aus $L(x)$

erzeugt wird:

c) Liegt $Q\bar{z}_{n+1} \bar{w}_i R$ ($1 \leq i \leq n$) vor, so ist nicht ausgeschlossen, daß die erzeugte Kette von Endsymbolen Anfangsstück eines Wortes aus $L(x)$ ist. Daher werden in diesem Fall lediglich alle verschiedenen Ketten $\kappa_i^j \in \Omega(\{a, b\})$ mit $|\kappa_i^j| \leq l_1$ und falls $|\kappa_i^j| = l_1$ Ketten $\kappa_i^j \neq x_1$ erzeugt: Es wird nach $Q\kappa_i^j \bar{z} R$ bzw. $Q\kappa_i^j R$ übergegangen.

d) Beim Vorliegen von $Q\bar{z}_{n+1} \bar{w}_i R$ ($1 \leq i \leq n$) kann die unter c) erwähnte Möglichkeit nicht auftreten, und es werden alle verschiedenen Ketten $\alpha_i^j \in \Omega(\{a, b\})$ mit $|\alpha_i^j| \leq l_1$ erzeugt: Es wird nach $Q\alpha_i^j \bar{z} R$ bzw. $Q\alpha_i^j R$ übergegangen.

e) Das Auftreten von $\bar{z}_{n+1}$ besagt allgemein, daß die Möglichkeit c) nicht mehr gegeben ist. Daher werden beim Vorliegen von $Q\bar{z}_{n+1} \bar{w}_i R$ alle verschiedenen Ketten $\alpha_i^j \in \Omega(\{a, b\})$ mit $|\alpha_i^j| \leq l_1$ erzeugt: Es wird nach $Q\alpha_i^j \bar{z}_{n+1} R$ bzw. $Q\alpha_i^j R$ übergegangen.

Um zuletzt noch alle Ketten der Form $\{ba^{\mu_1} \dots ba^{\mu_n} w^{\mu} /$
 $/ \mu \geq 1, 1 \leq i, \mu \leq n \text{ für } \mu=1, \dots, k, w^{\mu} \in \Omega(\{a, b\})$
 $\wedge |w^{\mu}| \equiv |x_{i_1} \dots x_{i_k}|_{\Lambda} w^{\mu} \neq x_{i_1} \dots x_{i_k}\}$ darzustellen, genügt es ersichtlicherweise das im Beweis von Lemma 7-1 eingeführte B-System leicht zu modifizieren.

Nach Satz 4-2 ist jede reguläre Sprache auch Puffersprache, und nach Satz 6-2 ist auch jede von einem B-System erzeugte Sprache Puffersprache. Mit Lemma 7-4 folgt dann daraus, daß die Menge $\bar{L}(x)$ eine Puffersprache darstellt.

Lemma 7-6: Es ist rekursiv unentscheidbar, ob gilt: $L(x) \cap L(y) = \emptyset$.
 Der Beweis verläuft analog zu dem von Satz 7-2 (a).

Mithilfe dieser Lemmata können wir den folgenden Satz beweisen:

Satz 7-3: Für nichtdeterministische Puffersprachen L_1 und L_2 sind die folgenden Probleme rekursiv unentscheidbar:

(a) Gilt $L_1 \subseteq L_2$?

(b) Gilt $L_1 = L_2$?

Bew.: Wir betrachten die Vereinigung $\bar{L}$ der Puffersprachen $\bar{L}(x)$ und $\bar{L}(y)$, die nach Lemma 7-4 wieder eine Puffersprache ist.

$$\begin{aligned}\bar{L} &:= \bar{L}(x) \cup \bar{L}(y) \\ &= ((\Omega(\{a,b\}) \setminus \{\Lambda\}) \setminus L(x)) \cup ((\Omega(\{a,b\}) \setminus \{\Lambda\}) \setminus L(y)) \\ &= (\Omega(\{a,b\}) \setminus \{\Lambda\}) \setminus (L(x) \cap L(y))\end{aligned}$$

Außerdem ist $\Omega(\{a,b\}) \setminus \{\Lambda\}$ als reguläre Sprache nach Satz 4-2 ebenfalls Puffersprache. Nach Lemma 7-6 ist es aber nicht entscheidbar, ob $L(x) \cap L(y) = \emptyset$ gilt. Daher ist es auch unentscheidbar, ob $(\Omega(\{a,b\}) \setminus \{\Lambda\}) \subseteq \bar{L}$ gilt, und da $\bar{L}$ genau dann gleich $\Omega(\{a,b\}) \setminus \{\Lambda\}$, wenn $L(x) \cap L(y) = \emptyset$, so ist auch die Gleichheit rekursiv unentscheidbar.

Die folgenden Ergebnisse fallen noch ab:

Satz 7-4: Für nichtdeterministische Puffersprachen L , kontextfreie Sprachen C und reguläre Sprachen R sind die folgenden Probleme rekursiv unentscheidbar:

(a) Gilt $L = C$?

(b) Gilt $L = R$?

Bew.: Nach Satz 7-3 ist es rekursiv unentscheidbar, ob $\bar{L} = \Omega(\{a,b\}) \setminus \{\Lambda\}$ gilt. Da letztere aber eine reguläre Sprache und also auch eine kontextfreie Sprache ist, folgt die Behauptung.

8. Eine Hierarchie von Pufferautomaten

In Analogie zu Arbeiten, die Turingmaschinen mit mehreren Arbeitsbändern zum Thema haben (s.z.B. ROSENBERG (1967)), werden wir im folgenden Automaten betrachten, die mit mehreren Puffern versehen sind und diese gleichzeitig verändern und ihr Verhalten von deren Inhalt abhängig machen. Diese n -Pufferautomaten, wie wir sie nennen wollen, werden wieder in ihrer Eigenschaft als erkennende Automaten beschrieben, die natürlich ebenfalls wieder in "Realzeit" arbeiten sollen, d.h., bei jedem Zustandsübergang muß ein neues Symbol der Eingabekette gelesen werden. Das Annehmen von Ketten wollen wir, etwas anders als zuvor, nicht mehr über das Leersein der Puffer definieren, sondern sie als angenommen bezeichnen, wenn sich der Automat nach dem Einlesen der Kette in einem markierten Zustand befindet. Ansonsten sind Begriffsbildungen, Bezeichnungen und Vorgehensweise im Kern die gleichen wie in den früheren Abschnitten, so daß wir uns oft mit Skizzierungen begnügen werden.

Nach den einführenden Definitionen werden wir den Zusammenhang mit den einfachen deterministischen Pufferautomaten aufzeigen und nachweisen, daß die Klasse der von n -Pufferautomaten angenommenen Sprachen eine Boolesche Algebra bildet und daß diese Sprachen hierarchisch gegliedert sind.

Def.: Ein deterministischer n -Pufferautomat ($n \geq 1$) ist ein Septupel $\mathcal{S}_P^{(n)} = (X, Z, P, B_2, d^{(n)}, z_0, F)$. Dabei sind X, Z und P endliche, nichtleere Mengen, $B_2 := \{b_{20}, b_{21}\}$, $F \subset Z$ und $d^{(n)}$ ist eine Abbildung von $Z \times X \times P^n$ in $Z \times P^n \times B_2^n$.¹⁾

Die Definitionen der Relation $\vdash_{\mathcal{S}_P^{(n)}}$, die die Arbeitsweise des n -Pufferautomaten zu beschreiben gestattet, des direkten Über-

1) A^n bezeichne das n -fache cartesische Produkt der Menge A .

gangs, des Übergangs und der Übergangsfolge wollen wir nicht explizit angeben, sondern auf die Analogie zum einfachen deterministischen Pufferautomaten hinweisen.

Ein Wort $w \in \Omega(X)$ der Länge k heißt vom deterministischen n -Pufferautomaten $\mathcal{L}_D^{(n)}$ angenommen, wenn vom Anfangszustand z_0 und leerem Puffer ausgehend die Übergangsfolge der Länge k zu einem markierten Zustand führt. Die Puffer können dabei zum Schluß -im Gegensatz zu früher- eine beliebige Inschrift haben. Die Menge der von einem deterministischen n -Pufferautomaten angenommenen Wörter wollen wir eine deterministische n -Puffersprache nennen.

Der Vergleich des einfachen deterministischen Pufferautomaten und des deterministischen 1-Pufferautomaten wird zeigen, daß der erste aufgrund der Forderung eines leeren Puffers eine geringere "Annahmekapazität" als der letztere besitzt:

Satz 8-1: (a) Zu jedem einfachen deterministischen Pufferautomaten $\mathcal{L}_D$ gibt es einen deterministischen 1-Pufferautomaten $\mathcal{L}_D^{(1)}$, so daß gilt: $T(\mathcal{L}_D) = T(\mathcal{L}_D^{(1)})$.

(b) Das Umgekehrte gilt nicht.

Bew.: (a) Der Automat $\mathcal{L}_D^{(1)}$ muß so konstruiert werden, daß er genau dann in einen markierten Zustand gelangt, wenn der Puffer von $\mathcal{L}_D$ leer ist. Da es in einem Pufferautomaten keine Möglichkeit gibt, abzufragen, ob sich noch genau ein Symbol im Puffer befindet, muß dieser Vorgang simuliert werden: Durch eine Erweiterung der Zustandsmenge gegenüber der von $\mathcal{L}_D$ wird es möglich, das jeweils zuletzt in $\mathcal{L}_D$ in den Puffer geschriebene Symbol durch Annehmen eines bestimmten Zustandes zu "speichern". Gelangt in $\mathcal{L}_D$ ein weiteres Symbol in den Puffer, so wird das gespei-

cherte Symbol in den Puffer von $\mathcal{L}_D^{(n)}$ eingeschrieben und das neue Symbol gespeichert. Ist dann der Puffer von $\mathcal{L}_D^{(n)}$ leer, -was sich ja feststellen läßt-, so wird, falls in $\mathcal{L}_D$ ein Symbol aus dem Puffer ausgelesen wird, in einen markierten Zustand übergegangen. Ist dann die Eingabekette noch nicht abgearbeitet und wird wieder ein Symbol in den Puffer von $\mathcal{L}_D$ geschrieben, so wird der markierte Zustand verlassen.

Gegeben sei ein einfacher deterministischer Pufferautomat

$\mathcal{L}_D = (X, Z, P, B_2, d, z_0)$. Dann wird folgendermaßen ein deterministischer 1-Pufferautomat $\mathcal{L}_D^{(n)} = (X^{(n)}, Z^{(n)}, P^{(n)}, B_2, d^{(n)}, z_0^{(n)}, F)$ konstruiert:

$X^{(n)} := X$; $Z^{(n)} := (Z \times \bar{P}) \cup \{(z_0, p_0), (\bar{z}, p_0)\}$, wobei $p_0 \notin P$ und $\bar{z} \notin Z$;
 $P^{(n)} := P$; $z_0^{(n)} := (z_0, p_0)$; $F := \{(z, \wedge) / (z, \wedge) \in Z \times \{\wedge\}\}$.

In Abhängigkeit von d wird $d^{(n)}$ gebildet:

(1) a) Für alle $x \in X$:

$$d^{(n)}((z_0, p_0), x, \wedge) = ((z', p'), \wedge, b_{21}),$$

falls $d(z_0, x, \wedge) = (z', p', b_{21})$, wobei $p' \in \bar{P}$

b) Für alle $x \in X$ und für alle $p \in P$:

$$d^{(n)}((z_0, p_0), x, p) = ((\bar{z}, p_0), \wedge, b_{20})$$

(2) Für alle $x \in X$ und für alle $p \in \bar{P}$:

$$d^{(n)}((\bar{z}, p_0), x, p) = ((\bar{z}, p_0), \wedge, b_{20})$$

(3) Für alle $x \in X$ und für alle $p \in P$:

$$d^{(n)}((z, \wedge), x, p) = ((\bar{z}, p_0), \wedge, b_{20})$$

(4) Für alle $z \in Z$ und für alle $x \in X$:

$$d^{(n)}((z, \wedge), x, \wedge) = ((z', p'), \wedge, b_{21}),$$

falls $d(z, x, \wedge) = (z', p', b_{21})$, wobei $p' \in \bar{P}$

(5) Für alle $(z, p) \in Z \times P$ und für alle $x \in X$:

$$d^{(n)}((z, p), x, \wedge) = ((z', p), \wedge, b_{20}),$$

falls $d(z, x, p) = (z', \wedge, b_{20})$

$$d^{(i)}((z,p),x,\wedge) = ((z',\wedge),\wedge,b_{20}),$$

$$\text{falls } d(z,x,p) = (z',\wedge,b_{21})$$

$$d^{(i)}((z,p),x,\wedge) = ((z',p'),p,b_{20}),$$

$$\text{falls } d(z,x,p) = (z',p',b_{20}), \text{ wobei } p' \neq \wedge$$

$$d^{(i)}((z,p),x,\wedge) = ((z',p'),\wedge,b_{20}),$$

$$\text{falls } d(z,x,p) = (z',p',b_{21}), \text{ wobei } p' \neq \wedge$$

(6) Für alle $(z,p) \in Z \times P$, für alle $x \in X$ und für alle $\bar{p} \in P$:

$$d^{(i)}((z,p),x,\bar{p}) = ((z',p),\wedge,b_{21}),$$

$$\text{falls } d(z,x,\bar{p}) = (z',\wedge,b_{21})$$

$$d^{(i)}((z,p),x,\bar{p}) = ((z',p),\wedge,b_{20}),$$

$$\text{falls } d(z,x,\bar{p}) = (z',\wedge,b_{20})$$

$$d^{(i)}((z,p),x,\bar{p}) = ((z',p'),p,b_{21}),$$

$$\text{falls } d(z,x,\bar{p}) = (z',p',b_{21}), \text{ wobei } p' \neq \wedge$$

$$d^{(i)}((z,p),x,\bar{p}) = ((z',p'),p,b_{20}),$$

$$\text{falls } d(z,x,\bar{p}) = (z',p',b_{20}), \text{ wobei } p' \neq \wedge$$

Durch vollständige Induktion über die Länge der Übergangsfolge ergibt sich dann sofort:

$$(z_0, ww', \wedge) \vdash_{\mathcal{Z}_D}^* \begin{cases} (z, w', \pi p') \\ (z, w', \wedge) \end{cases}$$

$$\Leftrightarrow ((z_0, p_0), ww', \wedge) \vdash_{\mathcal{Z}_D^{(i)}}^* \begin{cases} ((z, p'), w', \pi) \\ ((z, \wedge), w', \wedge) \end{cases}$$

wobei $w \in \Omega(X) \setminus \{\wedge\}$, $w' \in \Omega(X)$, $\pi \in \Omega(P)$, $p' \in P$.

D.f., $w \in T(\mathcal{Z}_D) \Leftrightarrow w \in T(\mathcal{Z}_D^{(i)})$.

(b) Am Beispiel der Menge $L = \{b^{k+1}a^k \mid k, l \in \mathbb{N}\}$ wollen wir den Teil (b) des Satzes beweisen. Im Beweis von Satz 5-4 wurde gezeigt, daß es keinen einfachen deterministischen Pufferautomaten gibt, der L annimmt. Für den folgenden deterministischen 1-Pufferautomaten $\mathcal{Z}_D^{(i)} = (X, Z, P, B_2, d^{(i)}, z_0, F)$ gilt jedoch, wie man leicht überlegt, $T(\mathcal{Z}_D^{(i)}) = L$.

$X := \{a, b\}$; $Z := \{z_0, z_1, z_2, z_3\}$; $P := \{p\}$; $F := \{z_2\}$ und $d^{(n)}$ ist folgendermaßen gegeben:

$$\begin{array}{ll} d^{(n)}(z_0, a, \wedge) = (z_3, p, b_{20}) & d^{(n)}(z_2, a, \wedge) = (z_3, p, b_{20}) \\ d^{(n)}(z_0, b, \wedge) = (z_1, \wedge, b_{20}) & d^{(n)}(z_2, b, \wedge) = (z_3, p, b_{20}) \\ d^{(n)}(z_0, a, p) = (z_3, p, b_{20}) & d^{(n)}(z_2, a, p) = (z_2, \wedge, b_{21}) \\ d^{(n)}(z_0, b, p) = (z_3, p, b_{20}) & d^{(n)}(z_2, b, p) = (z_3, p, b_{20}) \\ d^{(n)}(z_1, a, \wedge) = (z_3, p, b_{20}) & d^{(n)}(z_3, a, \wedge) = (z_3, p, b_{20}) \\ d^{(n)}(z_1, b, \wedge) = (z_1, p, b_{20}) & d^{(n)}(z_3, b, \wedge) = (z_3, p, b_{20}) \\ d^{(n)}(z_1, a, p) = (z_2, \wedge, b_{21}) & d^{(n)}(z_3, a, p) = (z_3, p, b_{20}) \\ d^{(n)}(z_1, b, p) = (z_1, p, b_{20}) & d^{(n)}(z_3, b, p) = (z_3, p, b_{20}) \end{array}$$

Bezüglich der Operationen mit deterministischen n-Puffersprachen erhalten wir anders als bei den deterministischen Puffersprachen (s. Abschnitt 5.) die folgende Aussage:

Satz 8-2: Die deterministischen n-Puffersprachen bilden bzgl. Vereinigung, Durchschnitt und Komplement (hinsichtlich $\Omega(X) \setminus \{\wedge\}$) eine Boolesche Algebra.

Bew.: Sei L_n eine deterministische n-Puffersprache. Dann gibt es nach Definition einen deterministischen n-Pufferautomaten $\mathcal{Z}_D^{(n)} = (X, Z, P, B_2, d^{(n)}, z_0, F)$, der L_n annimmt. Es ist offensichtlich, daß der deterministische n-Pufferautomat $\bar{\mathcal{Z}}_D^{(n)} = (X, Z, P, B_2, d^{(n)}, z_0, Z \setminus F)$ die Menge $(\Omega(X) \setminus \{\wedge\}) \setminus L_n$ annimmt.

Seien L_n und L_m zwei deterministische n- bzw. m-Puffersprachen über demselben Alphabet X und $\mathcal{Z}_D^{(n)} = (X, Z^{(n)}, P^{(n)}, B_2, d^{(n)}, z_0^{(n)}, F^{(n)})$ und $\mathcal{Z}_D^{(m)} = (X, Z^{(m)}, P^{(m)}, B_2, d^{(m)}, z_0^{(m)}, F^{(m)})$ deterministische n- bzw. m-Pufferautomaten, für die gilt: $L_n = T(\mathcal{Z}_D^{(n)})$ und $L_m = T(\mathcal{Z}_D^{(m)})$. Der deterministische $(n+m)$ -Pufferautomat $\mathcal{Z}_D^{(n+m)}$ wird folgendermaßen konstruiert:

$$\mathcal{L}_p^{(n+m)} = (X, Z^{(n)} \times Z^{(m)}, P^{(n)} \times P^{(m)}, B_2, d^{(n+m)}, (z_0^{(n)}, z_0^{(m)}), (P^{(n)} \times Z^{(m)}) \cup (Z^{(n)} \times P^{(m)})).$$

Dabei wird $d^{(n+m)}$ in naheliegender Weise so in Abhängigkeit von $d^{(n)}$ und $d^{(m)}$ konstruiert, daß $\mathcal{L}_p^{(n+m)}$ in den "ersten" n Puffern die Vorgehensweise von $\mathcal{L}_p^{(n)}$ und in den "letzten" m Puffern die von $\mathcal{L}_p^{(m)}$ simuliert. Ein Wort der Länge k wird offensichtlich von $\mathcal{L}_p^{(n+m)}$ genau dann angenommen, wenn es von einem der beiden Automaten $\mathcal{L}_p^{(n)}$ und $\mathcal{L}_p^{(m)}$ angenommen wird, d.h.

$$T(\mathcal{L}_p^{(n+m)}) = L_n \cup L_m.$$

Nach dem Satz von de Morgan folgt dann die Behauptung.

Es soll bemerkt werden, daß für festgehaltene n der obige Satz nicht mehr gültig bleibt.

Eine Aussage über den hierarchischen Aufbau der von Pufferautomaten mit mehreren Puffern angenommenen Sprachen macht der folgende Satz:

Satz 8-3: Die Klasse der von deterministischen n -Pufferautomaten ($n \in \mathbb{N}$) angenommenen Mengen ist echt in der Klasse der von deterministischen $(n+1)$ -Pufferautomaten angenommenen Mengen enthalten.

Bew.:¹⁾ Daß die Klasse der von deterministischen $(n+1)$ -Pufferautomaten angenommenen Mengen die Klasse der von deterministischen n -Pufferautomaten angenommenen Mengen enthält, ist nach Definition klar.

Die Echtheit der Enthaltenseinsbeziehung weisen wir für jedes n jeweils am Beispiel der Menge

$$L_n := \{u_0 c u_1 c \dots c u_n d u_k e \mid u_i \in \Omega(\{a, b\}) \setminus \{\Lambda\} \text{ für } i=0, \dots, n \text{ } 0 \leq k \leq n\}$$

nach, für die gilt:

(a) Es gibt einen deterministischen $(n+1)$ -Pufferauto-

¹⁾ Wir wollen den Beweis in weiten Teilen nur skizzieren, da für eine genaue Durchführung Definitionen benutzt werden müßten, die nicht von weiterem Interesse sind.

maten $\mathcal{L}_D^{(n+1)}$, so daß $L_n = T(\mathcal{L}_D^{(n+1)})$.

(b) Es gibt keinen deterministischen n-Pufferautomaten, der L_n annimmt.

Die Behauptung (a) ist jeweils unmittelbar einzusehen, so daß wir darauf nicht näher eingehen wollen.

Um die Behauptung (b) zu beweisen, werden wir die verschiedenen Fälle des Abspeicherns von Information in den Puffer untersuchen und zeigen, daß sich jedes Mal ein Widerspruch zur Annahme, daß es einen deterministischen n-Pufferautomaten $\mathcal{L}_D^{(n)}$ gibt, für den $T(\mathcal{L}_D^{(n)}) = L_n$ gilt, ergibt.

Für ein festes n ($n \geq 1$) nehmen wir an, es gebe einen deterministischen n-Pufferautomaten $\mathcal{L}_D^{(n)}$ mit $T(\mathcal{L}_D^{(n)}) = L_n$. Da L_n keine reguläre Sprache ist, muß von der Möglichkeit, Information zu speichern, Gebrauch gemacht werden. Zunächst zeigt man wie früher, daß sich der Automat $\mathcal{L}_D^{(n)}$ wegen seines deterministischen Verhaltens nach der Abarbeitung von $u_0 c \dots c u_n$ in einer durch diese Teilkette umkehrbar eindeutig bestimmten Konfiguration befinden muß. Den Begriff des Pufferbildes wollen wir wiederum in einer naheliegenden Erweiterung der Definition von Abschnitt 5. gebrauchen. Es ist wieder zu beachten, daß die Länge des Pufferbildes linear mit der Länge der Ketten wächst.

Es ergeben sich dann folgende Möglichkeiten des Abspeicherns des Pufferbildes von $u_0 c \dots c u_n$ und des anschließenden Vergleichens:

- ① a) Die Pufferbilder der Ketten u_i ($0 \leq i \leq n$) sollen in jeweils einem Puffer stehen. Daraus folgt aber,

daß die Pufferbilder π_l und π_m zweier Ketten u_l bzw. u_m ($0 \leq l, m \leq n$) mit $l < m$ in dieser Reihenfolge in den Puffer geschrieben werden.

Liegt dann eine Kette $u_0 c \dots c u_n$ vor mit $u_j c u_j$ für $j=0, \dots, m-1, m+1, \dots, n$ vor, so kann nicht festgestellt werden, ob u_m und u übereinstimmen.

Bei einer Vertauschung der Pufferbilder bleibt die analoge Argumentation für u_l gültig.

b) Auch im Fall, daß im Gegensatz zu a) das Pufferbild mindestens einer Kette "abwechselnd" (in einer beliebigen Weise) in mindestens zwei Puffer geschrieben wird, gilt das unter a) Gesagte.

c) Besteht das Pufferbild einer Kette aus mindestens zwei identischen Ketten in der entsprechenden Zahl von Puffern, so folgt daraus wie in a) der Widerspruch.

② Es bleibt noch die Möglichkeit zu diskutieren, durch entsprechende Codierung Pufferbilder zweier Ketten "übereinander" in einen Puffer zu schreiben, d.h., in einen Puffer eine Kette φ aus $\Omega(P \times P)$, wobei P das Pufferalphabet sei, aufzunehmen; dabei stellen die Kette, die aus den ersten Komponenten der Paare von φ durch Konkatenation gebildet ist, eine Codierung der Kette u_l und die aus den zweiten Komponenten gebildete Kette eine Codierung von u_m mit $l < m$ dar.

a) Da zunächst ein Pufferbild π_l von u_l in einen Puffer geschrieben wird, muß das Einschreiben der Codierung von u_m unter gleichzeitigem Auslesen von π_l geschehen. Da es aber in L_n Ketten gibt, für die

$|\pi_l| - |u_m| \gg \sum_{i=m+1}^n |u_i|$ gilt, läßt sich beim Abarbeiten

einer Kette $u_0 c \dots c u_n$ due mit $u \neq u_j$ für $j=0, \dots, m-1, m+1, \dots, n$ die Übereinstimmung von u_m und u nicht überprüfen.

b) Auch beim "Übereinanderschreiben" von mehr als zwei Ketten in einen Puffer oder von einer Kette in mindestens zwei Puffer bleiben die Überlegungen von a) gültig.

c) Das abwechselnde Einschreiben, dem Fall ① b) entsprechend, macht ebenfalls keine weiteren Untersuchungen notwendig.

Diese Argumentation läßt sich in der Feststellung zusammenfassen, daß für den deterministischen n -Pufferautomaten jeweils "zu spät" ersichtlich wird, welche Information im Puffer von Bedeutung war und die dadurch nicht rechtzeitig am Anfang eines Puffers zugreifbar war.

Wir haben damit auch das Ergebnis, daß es für jede natürliche Zahl n eine kontextsensitive, nicht kontextfreie Sprache gibt - nämlich L_n -, die von keinem deterministischen n -Pufferautomaten angenommen wird.

Mit denselben Worten wie oben zeigt man, daß es für jede natürliche Zahl n auch eine kontextfreie Sprache (allerdings eine nichtdeterministische kontextfreie Sprache) gibt, - nämlich

$L'_n := \{u_0 c \dots c u_n d u_k^R \mid u_1 \in \Omega(\{a, b\}) \setminus \{a\} \text{ für } i=0, \dots, n \wedge 0 \leq k \leq n\}$ -, die von keinem deterministischen n -Pufferautomaten angenommen wird.

Zu Entscheidbarkeitsfragen bei deterministischen n -Puffersprachen können wir folgendes sagen:

Da es zu jeder deterministischen Puffersprache einen deterministischen n -Pufferautomaten gibt, der sie annimmt, sind die in Satz 7-2 aufgeführten Probleme auch für deterministische n -Puffersprachen unentscheidbar.

Außerdem können wir noch eine Aussage über ein Problem machen, das für deterministische Puffersprachen offen bleiben mußte:

Satz 8-4: Für deterministische n -Puffersprachen L sind die folgenden Probleme rekursiv unentscheidbar:

(a) Ist L leer ?

(b) Enthält L unendlich viele Elemente ?

Der Beweis ergibt sich unmittelbar aus den Sätzen 8-1 und 8-2, da daraus folgt, daß die Menge $L_1 \cap L_2$ aus dem Beweis von Satz 7-2 eine deterministische n -Puffersprache ist, und aus Satz 7-2 (a) bzw. (b).

Literatur

- CHOMSKY, N. (1959), On certain formal properties of grammars.
Information and Control 2, 137-167
- DAVIS, M. (1958), Computability and Unsolvability.
McGraw-Hill, New York, London
- GINSBURG, S. (1966), The Mathematical Theory of Context-Free
Languages. McGraw-Hill, New York, London
- GINSBURG, S., ROSE, G. F. (1966), Preservation of languages by
transducers. Information and Control 9, 153-176
- HÄNDLER, W. (1966), Automatentheorie II (Vorlesungsausarbeitung).
Hannover
- HERMES, H. (1961), Aufzählbarkeit, Entscheidbarkeit, Berechenbarkeit.
Springer, Berlin, Göttingen, Heidelberg
- KURODA, S. Y. (1964), Classes of languages and linear-bounded
automata. Information and Control 7, 207-223
- LANDWEBER, P. S. (1963), Three theorems on phrase structure
grammars of type 1. Information and Control 6, 131-136
- LANDWEBER, P. S. (1964), Decision problems of phrase-structure
grammars. IEEE Trans. on Electronic Computers EC-13, 354-362
- POST, E. L. (1946), A variant of a recursively unsolvable problem.
Bull. Am. Math. Soc. 52, 264-268
- RABIN, M. O. (1963), Real time computation.
Israel Journal of Math. 1, 203-211
- RABIN, M. O., SCOTT, D. (1959), Finite automata and their decision
problems. IBM J. Res. Dev. 3, 114-125

ROSENBERG, A.L. (1967), Real-time definable languages.

Journal of the ACM 14, 645-662

SCHMITT, A. (1966), Theorie der nichtdeterministischen und unvollständigen Mealy-Automaten. Diss., Hannover

SCHNEIDER, H.J. (1965), Ein formales Verfahren zur maschinellen Sprachanalyse. Diss.; Hannover

SHEPHERDSON, J.C., STURGIS, H.E. (1963), Computability of recursive functions. Journal of the ACM 10, 217-255

Der Verfasser schuldet Dank Herrn Professor Dr. W. Händler für die Förderung dieser Arbeit und Herrn Dr. A. Schmitt und Herrn Dr. H. J. Schneider für viele Anregungen und wertvolle Hinweise. Allen Kollegen der Arbeitsgruppe "Automatentheorie und Mathematische Linguistik" des Instituts für Mathematische Maschinen und Datenverarbeitung sei für ihre Diskussionsbemerkungen gedankt.

Lebenslauf

Geboren am 1. November 1939 in Braubach/Rh. als Sohn des Reichsbahninspektors Gustav Vollmar und seiner Frau Anna, geb. Rothhaar.

Kindheitsjahre in Elsaß-Lothringen und in Rosenkopf/Pfalz. Besuch der Volksschule Rosenkopf 1945-46, der Volksschule Zweibrücken 1946-49, der Oberrealschule in Zweibrücken 1949-50, des Staatlichen Naturwissenschaftlichen Gymnasiums in Kaiserslautern 1950-58. Nach dem Abitur im Februar 1958 Studium der Mathematik an der Universität Heidelberg 1958-59 und an der Universität des Saarlandes in Saarbrücken 1959-64. Nach dem Diplom seit Februar 1965 als wissenschaftlicher Mitarbeiter von Herrn Professor Dr. W. Händler tätig, zuerst am Lehrstuhl für Elektronische Rechenanlagen der Technischen Hochschule Hannover und seit dem Herbst 1966 am Institut für Mathematische Maschinen und Datenverarbeitung der Universität Erlangen-Nürnberg.

Offsetdruck-Fotodruck

J. Hogl, 8520 Erlangen Hauptstr. 109, (Nähe Martin-Luther-Platz) Tel. 22990
